

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Strategie ve fotbale robotů

Strategies in Robot Soccer

2012

Bc. Miroslav Osmančík

Zadání diplomové práce

Student:

Bc. Miroslav Osmančík

Studijní program:

N2647 Informační a komunikační technologie

Studijní obor:

2612T025 Informatika a výpočetní technika

Téma:

Strategie ve fotbale robotů

Strategies in Robot Soccer

Zásady pro vypracování:

Robotický fotbal je zajímavou oblastí k řešení problémů různých disciplín: zpracování obrazu, robotiky, umělé inteligence, atd. V rámci diplomové práce se student nejprve seznámí s hrou fotbal robotů a s výsledky, kterých bylo dosud dosaženo na katedře informatiky v dané oblasti. Student se dále zaměří na oblast strategií a taktik, zejména na detekci specifických strategií a jejich využití ve hře.

1. Seznamte se s hrou fotbal robotů.
2. Seznamte se s diplomovými pracemi z oblasti fotbalu robotů svých předchůdců a dalšími pracemi oborové katedry.
3. Prostudujte existující řešení z pohledu strategií a taktik.
4. Definujte typy strategií, které mohou být z pohledu hry důležitější než ostatní.
5. Navrhněte a naimplementujte algoritmy pro detekci, výběr a využití specifických strategií.
6. Proveďte experimenty a vyhodnoťte jejich výsledky.

Seznam doporučené odborné literatury:

1. Sammut C. Robot soccer. Wiley Interdisciplinary Reviews: Cognitive Science, Volume 1, Issue 6, pages 824–833, November/December 2010.
2. Diplomové a baklářské práce FEI VŠB - TUO týkající se fotbalu robotů.
3. Dále dle pokynů vedoucího práce.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **RNDr. Eliška Ochodková, Ph.D.**

Datum zadání: 18.11.2011

Datum odevzdání: 04.05.2012



doc. Dr. Ing. Eduard Sojka
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Souhlasím se zveřejněním této diplomové práce dle požadavků čl. 26, odst. 9 *Studijního a zkušebního řádu pro studium v magisterských programech VŠB-TU Ostrava*.

V Ostravě 1. května 2012

Miroslav Bunc
.....

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 1. května 2012

Miroslav Bunc
.....

Rád bych na tomto místě poděkoval všem, kteří mi s prací pomohli. Zvláštní poděkování patří vedoucí mé práce RNDr. Elišce Ochodkové, Ph.D., která věnovala spoustu času a energie, aby mi pomohla s řešením různých úskalí při vypracování praktické části a psaní práce. Poděkování patří také Ing. Janu Martinovičovi, Ph.D. a Ing. Václavu Svatoňovi za věcné náměty a připomínky v průběhu zpracování praktické části.

Abstrakt

Tato práce se zabývá optimalizací strategií pro 2D simulátor fotbalu robotů, který je vyvíjen na katedře informatiky. Teoretická část práce popisuje základní strukturu simulátoru, jeho využití a dosavadní vývoj problematiky strategií týmu. Práce se zejména zaměřuje na tři typy speciálních pravidel - cyklická, hub a útočná pravidla. Práce obsahuje teoretický návrh optimalizace těchto pravidel.

Praktická část se zaměřuje na implementaci optimalizačních algoritmů do simulátoru a ověřuje teorie navržené pro zlepšení strategií ve hře. Kód je napsán v jazyce C#. Cílem je odstranění nebo nahrazení nevhodných a nadbytečných pravidel ve strategii týmu a tím docílení větší úspěšnosti týmu v zápasech s různými soupeři. Po implementaci byly provedeny experimenty a testovací zápasy s různými strategiemi. Na závěr jsou v práci uvedeny jejich výsledky a zhodnocení.

Klíčová slova: strategie, simulátor, fotbal, robot, pravidlo, optimalizace, fotbal robotů, hub, cyklus, útok

Abstract

This thesis deals with an optimization of strategies in 2D robot soccer simulator which is being developed at the Department of Computer Science. The basic structure of the simulator, its current use and development of team strategies are described in the theoretical part of the thesis. It especially concentrates on three types of special rules - hub, loop and offensive rules. The thesis also includes theoretical design of the optimization of these special rules.

The practical part is focused on the implementation of optimization algorithms in the simulator and verifies the theories developed to improve the strategies. The code is written in C#. The aim is to remove or replace inappropriate or redundant rules in the strategy and thereby achieve the greater success of our robot team in soccer matches with various opponents. The experiments and test matches between different strategies were performed after the implementation. There are described the results and evaluation at the end.

Keywords: strategy, simulator, soccer, robot, rule, optimization, robot soccer, hub, loop, offensive

Seznam použitých zkratek a symbolů

FIFA	– Federation International Football Association
FIRA	– Federation of International Robotsoccer Association
PC	– Personal computer - osobní počítač
GUI	– Graphical User Interface - grafické uživatelské rozhraní

Obsah

1 Úvod	9
2 Fotbal robotů	11
2.1 Motivace	11
2.2 RoboCup	11
2.3 MiroSot	12
3 Simulátor	13
3.1 Definice pojmů	13
3.2 Stávající řešení	14
3.3 Strategy Creator	16
4 Strategie	19
4.1 Stávající řešení	19
4.2 Teoretický základ detekce speciálních pravidel ve strategii	19
5 Praktická implementace detekce speciálních pravidel	29
5.1 Cyklická pravidla	29
5.2 Hub pravidla	30
5.3 Útočná pravidla	33
6 Experimenty	35
6.1 Analýza cyklických pravidel	35
6.2 Využití detekce hub pravidel	37
6.3 Experimenty s generováním ofenzivních pravidel	39
6.4 Experimenty s kombinací optimalizací	41
7 Závěr	43
8 Literatura	45
Přílohy	45
A Obsah CD	47

Seznam tabulek

1	Výsledky zápasů před a po analýze cyklických pravidel	36
2	Výsledky zápasů před a po analýze hub pravidel	38
3	Výsledky zápasů před a po využití generování ofenzivních pravidel . . .	40
4	Výsledky zápasů s kombinací optimalizací	42

Seznam obrázků

1	RoboCup Germany Open 2010 [10]	12
2	Abstrakce hracího pole na gridy	14
3	Hlavní okno simulátoru	15
4	Strategy Creator	17
5	Strategy Creator - hlavička strategie	17
6	Cyklický pohyb hráčů na hřišti	21
7	Znázornění vlastních čísel v komplexní rovině	22
8	Orientovaný graf strategie s huby	24
9	Orientovaný graf strategie s přidanými pravidly GOAL a START	25
10	Graf strategie z výpisu č. 4 po nahrazení hubů odpovídající výpisu č. 5	28
11	Screenshot implementace analýzy cyklických pravidel v simulátoru	30
12	Screenshot implementace analýzy hub pravidel v simulátoru	32
13	Screenshot implementace generování útočných pravidel v simulátoru	34
14	Screenshot převodu pravidla na jiný grid	34

Seznam výpisů zdrojového kódu

1	Ukázka logovacího souboru - prvních 5 tiků hry	15
2	Ukázka dvou pravidel zapsaných v souboru strategie.strg	16
3	Výpis strategie s cyklickými pravidly	20
4	Ukázková strategie s hub pravidly	23
5	Upravená strategie po hub detekci	32

1 Úvod

Práce se zaměřuje na oblast robotiky. V současnosti se robotika a umělá inteligence dostává čím dál více do životů obyčejných lidí. Přístroje jako kávovar nebo kuchyňský robot jsou ve své podstatě roboti, kteří nám usnadňují práci. Umělá inteligence hraje velkou roli také v oblasti vesmírného výzkumu nebo zdravotnictví.

Tématem robotiky se zabýval již Karel Čapek na počátku 20. století ve své knize R.U.R., kde se vyskytuje značná kritika k přeceňování techniky a obava o nadvládu robotů a osud lidstva. Právě díky této hře se slovo „robot“ rozšířilo po celém světě. Autorství je připisováno Josefu Čapkovi, bratru Karla Čapka. Slovo „robot“ je slovanského původu a vzniklo ze slovesa „robotovat“ (pracovat).

Fotbal robotů na první pohled vypadá jako zábavná aplikace na hraní. Stává se ovšem celosvětově uznávaným vědecko-technickým měřením úrovně vyspělosti. Na jedné straně může být fotbal robotů opravdu hrou pro neprogramátory. Jeho důležitost ovšem tkví v integraci mnoha aspektů umělé inteligence. Cílem je vytvořit robota, který je schopen vykonávat nějakou činnost komplexně v dynamicky měnícím se prostředí. Rozhodování robota a jeho míra autonomie hraje roli v jeho chování. Robot musí pomocí senzorů vnímat okolní prostředí a reagovat na vnější podněty. Takovým podnětem může být kop do míče, běh na branku, gól, objetí soupeře, vyhnutí se mantinelu a další.

Tato práce je zaměřena na simulátor fotbalu robotů. Cílem je prostudování problematiky strategií ve hře a následné navržení a implementace optimalizačních algoritmů pro speciální pravidla ve strategii.

V druhé kapitole se budeme podrobněji věnovat obecnému fotbalu robotů. Zmíníme soutěže ve fotbale robotů a detailnější význam fotbalu robotů.

Třetí kapitola se již zaměřuje na samotný simulátor fotbalu robotů, do kterého je implementována praktická část této práce. Jelikož tato práce nevznikala od nuly, je zde popsán stávající simulátor, jeho funkce a architektura. Ve zvláštní podkapitole se zaměříme na tzv. *StrategyCreator* - aplikaci, ve které si každý může vytvořit strategii pro svůj fotbalový tým.

Následující kapitola se věnuje teoretickému významu a detekci speciálních typů pravidel, která se mohou ve strategiích vyskytovat a mít pro ně důležitější význam než ostatní pravidla. Zaměříme se na rozbor a detekci cyklických pravidel, útočných pravidel a tzv. hub pravidel.

Pátá kapitola popisuje praktickou implementaci navržených optimalizací do simulátoru. Popisuje třídu *SG.StrategyOptimization*, která obsahuje metody pro detekci všech tří speciálních typů pravidel ve strategiích.

V šesté kapitole jsou uvedeny různé experimentální hry, které byly odehrány před a po aplikaci detekce speciálních typů pravidel a následné zhodnocení jejich úspěšnosti.

V závěru práce je shrnut význam a úspěšnost detekce speciálních strategií. Dále zde jsou uvedeny náměty pro další práci a vylepšení simulátoru.

2 Fotbal robotů

Fotbal robotů se v posledních letech stal zajímavou a oblíbenou strukturou pro řešení problémů v různých oblastech. Odvětví umělé inteligence, autonomie více robotů, zpracovávání obrazu, řídicí techniky a další problémy se vyskytují právě v rámci robotického fotbalu. Cílem je co nejvíce přiblížit fotbal robotů tomu skutečnému fotbalu, který všichni známe.

Fotbalový tým robotů je podle odborníků vizitkou vysokých škol v oblasti robotiky a umělé inteligence. Právě mistrovství robotického fotbalu se většinou zúčastňují týmy z vysokých škol.

2.1 Motivace

V posledních letech se právě soutěže staly populárními v robotickém výzkumu. Přináší objektivní srovnání různých vědeckých postupů a poskytují jakousi metodu porovnávání úrovně výzkumu. Programátoři musí své systémy nastavit tak, aby jejich výkon a funkce byly zachovány i mimo laboratorní podmínky. Fotbal robotů se hraje v podmínkách, které určuje organizátor a ty jsou nezávislé na týmu.

RoboCup¹ je jednou z nejvýznamnějších organizací zabývajících se pořádáním soutěží v robotice - nejen ve fotbale robotů. Další významnou organizací je federace mezinárodních spolků robotického fotbalu FIRA² (založena 1995 v Jižní Koreji), která je populární zejména v Asii.

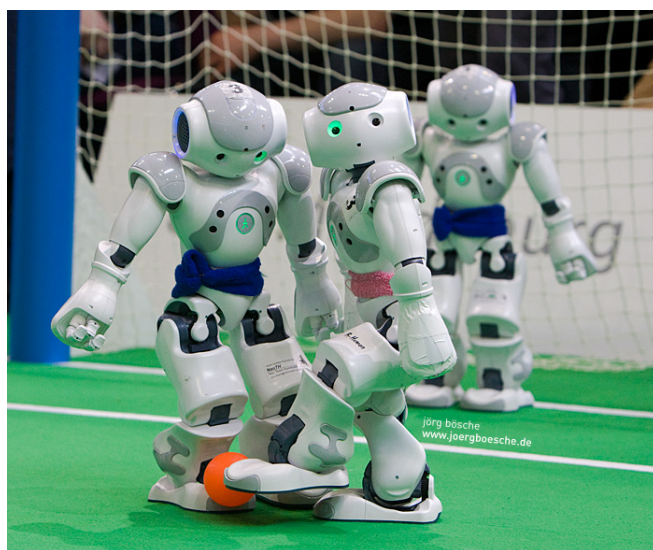
2.2 RoboCup

Jedna z největších mezinárodních soutěží v robotickém fotbale je RoboCup. Robot - hráč musí mít následující schopnosti:

- Robot musí být schopný vnímat okolní prostředí a tudíž se mu i přizpůsobit.
- Robot musí být schopen zpracovávat informace ze svých senzorů a tím si modelovat jakýsi svůj svět s vysokou přesností.
- Schopnost reagovat na aktuální změny okolí robota a na základě změn vykonávat určitou činnost.
- Přesné řízení motorů a jiných pohybových aparátů, aby byla daná činnost vykonána správně.
- Schopnost komunikace s ostatními hráči v týmu a vzájemná spolupráce při hře - využití strategií a taktik týmů.
- Učení - robot může mít schopnost naučit se kalibrovat senzory v nových okolních podmínkách nebo některé týmy využívají samoučení k kopírování soupeřovy strategie během hry a zakomponují vhodná pravidla do své hry.

¹<http://www.robocup.org>

²<http://www.fira.net>



Obrázek 1: RoboCup Germany Open 2010 [10]

RoboCup se skládá z několika lig podle typů robotů a cílového úkolu, případně podle věku programátorů (RoboCup Junior). Mimo robotický fotbal jsou zde ligy i pro výzkum pátracích a záchranných robotů (RoboCup Rescue). Některé ligy jsou vyhrazeny pouze pro simulátory robotů.

Cílem RoboCupu je propagace výzkumu a vývoje robotických aplikací. Vizí do budoucna, nosným cílem, RoboCupu je do roku 2050 sestavit takový tým plně autonomních humanoidních robotů, kteří budou hrát fotbal dle oficiálních pravidel FIFA a vyhrají nad vítězem posledního mistrovství světa ve fotbale.

Soccer Simulation League je jedna z nejstarších lig v RoboCupu. Liga simulátorů se zaměřuje na umělou inteligenci a týmové strategie. Nezávisle pohybující se hráči hrají fotbal ve virtuálním prostředí počítače. Tato liga má dvě podkategorie - 2D a 3D simulace.

2.3 MiroSot

Micro Robot World Cup Soccer (MiroSot³) je jedním z turnajů organizovaných již zmíněnou federací FIRA. Robot v tomto turnaji musí být menší než 7,5 x 7,5 x 7,5 cm, takže se nejedná o humanoidního robota, ale robota ve tvaru krychle. Je dána přesná velikost hřiště a počet hráčů. Na katedře informatiky se již pracovalo a stále pracuje na reálném fotbalu robotů v této formě.

Jedním z turnajů je také SimuroSot, což je turnaj robotických týmů v simulátoru. V návaznosti na tento turnaj se na katedře informatiky začalo pracovat i na simulátoru fotbalu robotů, který je platformou pro testování strategií týmů, které mohou být následně použité při hře s reálnými roboty. Vzhledem k tomu, že v reálném fotbale jsou roboti ve tvaru krychlí, tak v našem simulátoru jsou roboti reprezentováni čtverečky.

³<http://www.fira.net/?mid=mirosot>

3 Simulátor

Od roku 2007 se na katedře informatiky vyvíjí 2D simulátor fotbalu robotů. Pracovalo na něm již mnoho studentů v rámci svého bakalářského, magisterského nebo doktorského studia. Tento simulátor již také prošel mnoha změnami. V následující kapitole si představíme stav a funkce simulátoru, které byly aktuální v roce 2011 a 2012.

3.1 Definice pojmů

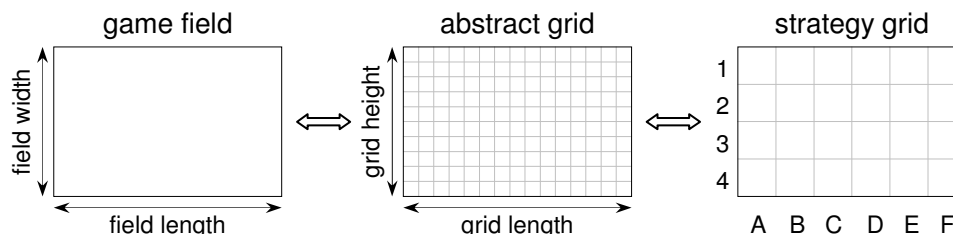
Před popisem postupů stávajícího stavu simulátoru a implementace je třeba ujasnit pojmy a termíny, které se budou v práci vyskytovat. Terminologie v oblasti strategií a taktik ve fotbale robotů bývá rozdílná v různých zdrojích. Pro naši práci a náš simulátor si definujeme následující pojmy:

- **Taktika**, jak ji budeme chápat pro tento text a práci v simulátoru, definuje chování robota (hráče) na malém úseku hřiště. Řeší dílčí problémy pro splnění dané strategie. Příkladem může být odebrání míče soupeři, vyhnutí se protihráči, řešení pohybu na pomezí čáře hřiště a pod. Taktika se váže na konkrétního hráče, ne na celý tým.
- **Pravidlo** - je základní prvek strategie. Pravidlo je definováno identifikátorem - číslem pravidla. Pravidlu můžeme přidělit typ (ofenzivní nebo defenzivní) a jméno. Skládá se ze čtyř dalších prvků - hodnoty *.Mine*, *.Oppnt*, *.Ball* a *.Move*. Všechny hodnoty jsou v gridových souřadnicích. *.Mine* udává výchozí pozici mých hráčů na hřišti, *.Oppnt* udává výchozí pozici protihráčů (mimo brankáře), *.Ball* udává pozici míče. *.Move* udává pozici mých hráčů (mimo brankáře), na kterou se mají přesunout pokud nastane postavení robotů a míče na hřišti dané hodnotami *.Mine*, *.Oppnt* a *.Ball*.

Vzhledem k tomu, že hra je dynamická a mění se v čase, pravidla se nemusí v průběhu hry dokončit a často se vybere již jiné, lepší pravidlo ze strategie k dalšímu postupu týmu.

- **Strategie** - slovo pocházející z řeckého *strategos* = generál, *stratos* = vojsko, výprava a *agein* = vést. Je to dlouhodobý plán činností zaměřený na dosažení nějakého cíle [11]. V našem simulátoru strategie představuje soubor pravidel, podle kterých se tým robotů řídí v průběhu hry. Strategie týmu je vytvářena předem, může být ovšem modifikována při přerušení hry. Pojem strategie se týká celého týmu. Jeden tým má jednu strategii obsahující různá pravidla. Strategii lze vytvořit přímo v simulátoru a následně je uložena do souboru s příponou *.strg*.
- **Reálné souřadnice** - vyjadřují x a y souřadnici pozice robota nebo míče v souřadnicovém systému na základě pixelů.
- **Grid** - sektor, který obsahuje danou množinu reálných bodů (pixelů). Gridy vznikly na základě zjednodušení hracího pole právě pro potřebu realizace strategií týmů.

Gridy tvoří síť, která pokrývá celé hrací pole. Grafické znázornění abstrakce hracího pole na gridy je znázorněno na obrázku č. 2.



Obrázek 2: Abstrakce hracího pole na gridy

- **Gridové souřadnice** - vyjadřují x a y souřadnici pozice robota nebo míče systému na základě gridů. Počet gridů do šířky i délky je možno měnit, podle toho, jak hustou síť gridů potřebujeme. Standardně je v simulátoru nastaven grid 6 x 4. Gridové souřadnice mají počátek v levém horním rohu hracího pole. Souřadnice x je počet gridů zleva doprava počínaje 1 a y souřadnice gridu je počet gridů shora dolů počínaje 1.

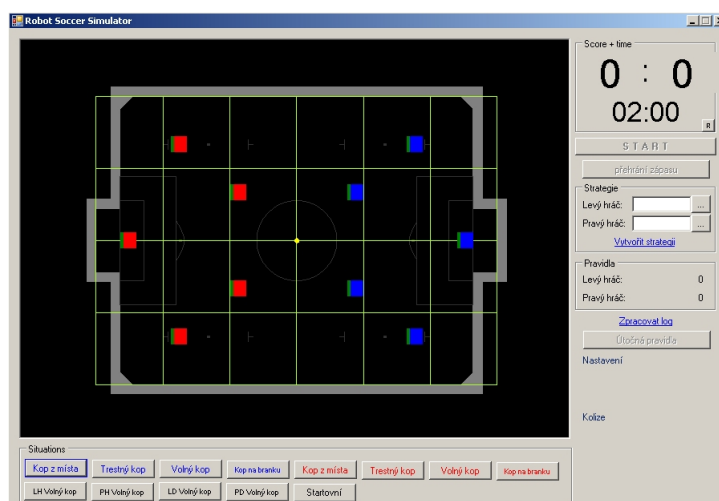
3.2 Stávající řešení

Hlavní funkcí simulátoru je co nejlépe simulovat hru fotbalu robotů. Simulátor obsahuje virtuální hrací plochu a pro každý tým 5 hráčů včetně brankáře. Hrací plocha je rozdělena do tzv. gridových souřadnic. Simulátor zaznamenává čas hry a skóre. Každý z hráčů si pro svůj tým může vybrat strategii a podle té hrát.

Stále se pracuje na implementaci automatického rozhodčího, který by hru reguloval a hlídal. Z toho důvodu někdy dochází k zablokování hráčů na jedné pozici. Nejsou řešeny ani jiné přestupky jako je ofsajd, vhazování, pokud míč přešel pomezní čáru, rohový kop, kop od branky a další. Tyto situace stále musí řešit lidský mozek - fyzický rozhodčí sledující hru. Pro tyto případy jsou v simulátoru přednastavené pozice hráčů pro různé speciální situace - kop z místa, trestný kop, kop na branku, startovní pozice obou družstev a další. Detailnější informace o skladbě simulátoru jsou popsány také v práci Lucie Žolté [5].

Interně jsou v simulátoru zakomponovány základní taktiky pro pohyb hráčů a jejich reakce. Hřiště je rozděleno na 4 základní sektory - brankoviště, defensivní území, ofenzivní území a okraje hřiště (mantinely - pomezní čáry). Podle toho, ve které části se hráč aktuálně nachází, vybírá taktiku dle které se bude chovat. Více o taktikách se dočtete v práci Sebastiana Balona [6].

Každá hra se loguje do tzv. *logovacích souborů*. V průměru proběhne v jedné sekundě asi 10 tiků hry. Standardní doba hry je 2 minuty, takže logovací soubor obsahuje více než 1200 řádků. Tato data se zapisují do textových souborů s příponou *.log*. Logování je zmíněno, protože hraje důležitou roli při analýze strategií. Z těchto souborů lze vyčíst, kdy bylo dané pravidlo použito, s jakou četností a naopak lze zjistit, která pravidla



Obrázek 3: Hlavní okno simulátoru

jsou ve strategii k ničemu, protože nejsou využívány. Ve výpisu 1 můžete vidět ukázkou logu prvních pěti tiků hry. První dvě hodnoty jsou čísla použitých pravidel levého a pravého týmu pro daný tik. Následující hodnoty jsou souřadnice pozic míče a všech robotů na hřišti vždy pro každého postupně v reálných souřadnicích a pak také v gridových souřadnicích. Poslední hodnota oddělená středníkem je dosažené skóre hry *levý tým : pravý tým*.

```
leftRule rightRule ; ball . real . x ; ball . real . y ; ball . grid . x ; ball . grid . y ; lr0 . real . x ; lr0 . real . y ; lr0 . grid . x ; lr0 . grid . y ; lr1 . real . x ; lr1 . real . y ; lr1 . grid . x ; lr1 . grid . y ; lr2 . real . x ; lr2 . real . y ; lr2 . grid . x ; lr2 . grid . y ; lr3 . real . x ; lr3 . real . y ; lr3 . grid . x ; lr3 . grid . y ; lr4 . real . x ; lr4 . real . y ; lr4 . grid . x ; lr4 . grid . y ; rr0 . real . x ; rr0 . real . y ; rr0 . grid . x ; rr0 . grid . y ; rr1 . real . x ; rr1 . real . y ; rr1 . grid . x ; rr1 . grid . y ; rr2 . real . x ; rr2 . real . y ; rr2 . grid . x ; rr2 . grid . y ; rr3 . real . x ; rr3 . real . y ; rr3 . grid . x ; rr3 . grid . y ; rr4 . real . x ; rr4 . real . y ; rr4 . grid . x ; rr4 . grid . y ;
```

```
13 1; 0;0;4;3;-36,67;-30;3;2;-36,67;30;3;3;-73,33;-60;2;1;-73,33;60;2;4;-105;0;1;3;36,67;-30;4;2;36,67;30;4;3;73,33;-60;5;1;73,33;60;5;4;105;0;6;3;0;0;
13 1; 0;0;4;3;-34,72;-28,4;3;2;-35,3;32,12;3;3;-70,82;-60,14;2;1;-72,46;62,36;2;4;-105;0;1;3;34,72;-28,4;4;2;34,17;29,67;4;3;73,01;-57,5;5;1;70,82;60,14;5;4;105;0;6;3;0;0;
13 1; 0;0;4;3;-31,93;-26,12;3;2;-33,34;35,14;3;3;-67,22;-60,34;2;1;-71,22;65,74;2;4;-105;0;1;3;31,93;-26,12;4;2;30,6;29,21;4;3;72,54;-53,93;5;1;67,22;60,34;5;4;105;0;6;3;0;0;
13 1; 0;0;4;3;-29,14;-23,84;3;2;-31,38;38,16;3;3;-63,63;-60,54;2;1;-69,97;69,12;2;4;-105;0;1;3;29,14;-23,84;4;2;27,03;28,74;4;3;72,07;-50,36;5;1;63,63;60,54;5;4;105;0;6;3;0;0;
13 1; 0;0;4;3;-26,36;-21,57;3;2;-29,43;41,18;3;3;-60,03;-60,73;2;1;-68,73;72,5;2;4;-105;0;1;3;26,36;-21,57;4;2;23,46;28,28;4;3;71,6;-46,79;5;1;60,03;60,73;5;4;105;0;6;3;0;0;
```

Výpis 1: Ukázkou logovacího souboru - prvních 5 tiků hry

Po vybrání strategií pro oba týmy může být spuštěn zápas. Standardně je nastaven zápas probíhající 2 minuty. Lze ho kdykoliv přerušit, zrušit nebo resetovat - uvést hráče do základního postavení. Při přerušení hry lze také uplynulou hru zpětně přehrát. Všechny tyto volby nabízí hlavní okno simulátoru. Při spuštění hry se doporučuje mít vypnutý

rezidentní štít antivirového programu, který máte na svém PC. Pokud necháme štít zapnutý, hra se také odehraje, ale může se stát, že bude „trhaná“ a proto i její výsledek nemusí být relevantní. Tento problém je způsoben stálou kontrolou souborů rezidentním štítem v operačním systému.

3.3 Strategy Creator

Strategy Creator je jednou z komponent simulátoru, ve které si může i neprogramátor vytvořit svou herní strategii pro tým. Po spuštění simulátoru najdeme v levé části okna odkaz „Vytvořit strategii“, přes který se nám otevře Strategy Creator - viz obrázek 4. GUI Strategy Creatoru je docela uživatelsky přívětivé. Intuitivně si může uživatel vytvořit novou strategii tím, že na hracím poli posouvá své a protihráčovy roboty a změnou jejich pozic vytvoří pravidla, která pak uloží do strategie. Tato strategie se uloží do textového souboru s příponou *.strg*. Ukázkou zapsaných pravidel v souboru strategie můžete vidět ve výpisu 2. Jsou zde 2 pravidla. Uživatel může kdykoliv tuto strategii otevřít, měnit ji, optimalizovat a podobně.

První řádek je označen slovem *.Rule*, následuje jedinečné číslo pravidla, písmeno *o* nebo *d* označující typ pravidla (ofensivní nebo defenzivní) a poslední položkou je název pravidla. Následující 4 řádky jsou uvozeny texty *.Mine*, *.Oppnt*, *.Ball* a *.Move* a za nimi vždy následují gridové souřadnice všech robotů na hřišti s výjimkou brankáře. Za textem *.Ball* následuje jen jedna souřadnice - pozice míče na hřišti. Pokud jsou naši roboti na hřišti rozestavení v situaci dané souřadnicemi *.Mine*, soupeřovi roboti v rozestavení *.Oppnt* a míč na pozici *.Ball*, pak se naši roboti začnou přesouvat na pozice dané souřadnicemi *.Move*. Stejný zápis můžeme vidět i pro pravidlo číslo 2, ale již s jinými souřadnicemi.

```
.Rule 1 o Name
.Mine 6,4 3,3 4,2 4,3
.Oppnt 5,4 4,3 5,2 4,3
.Ball 4,3
.Move 1,3 1,3 2,2 3,4

.Rule 2 o Name
.Mine 2,3 1,1 2,2 4,4
.Oppnt 4,4 6,2 6,3 5,2
.Ball 6,2
.Move 4,4 6,4 3,2 3,1
```

Výpis 2: Ukázka dvou pravidel zapsaných v souboru strategie.strg

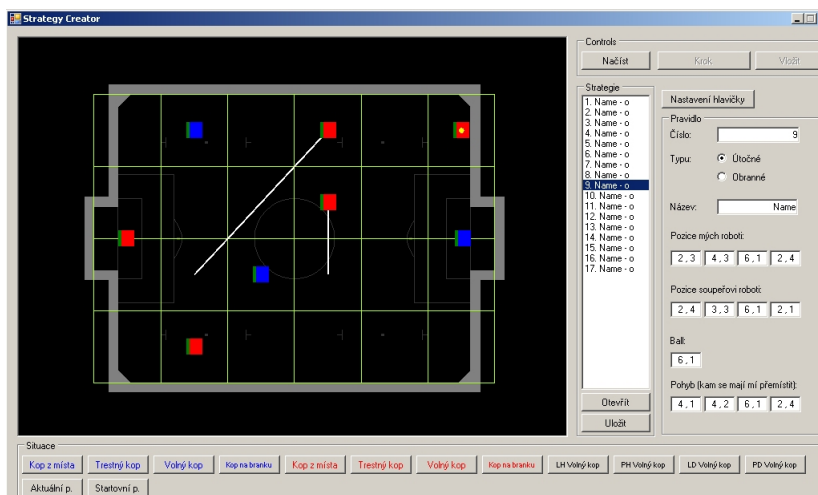
Postavení hráčů a tým i pravidla strategie se vztahují ke gridovým souřadnicím. Hustotu gridové sítě můžeme zvolit v hlavičce strategie.

Doporučuje se nastavení např. 6x4 tak, aby se tvar gridu blížil čtverci. Vzdálenosti mezi gridy v různých směrech se tak stanou podobné a pohyb robotů bude přesnější. Toho bychom nedosáhli, kdyby byl grid např. protáhlý obdélník.

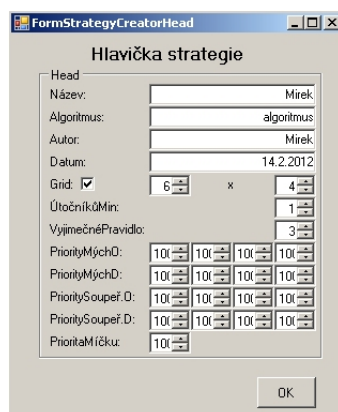
Během vytváření strategie, je možné každému pravidlu nastavit jeho typ - utočné nebo obranné. Simulátor tuto volbu zohledňuje při výběru pravidla ze strategie v dané herní situaci.

Pro každou strategii je možné také definovat tzv. *hlavičku* - viz obrázek 5. V hlavičce strategie můžeme zadat obecné informace jako je název strategie, její autor, algoritmus, datum vytvoření. Důležitým nastavením může být již zmiňovaná hustota gridové sítě. Vzhledem k tomu, že pravděpodobnost toho, že nastane přesně námi nadefinovaná situace v pravidlu je nízká, je třeba říct do jaké míry musí reálná situace odpovídat našemu nadefinovanému postavení hráčů. Toto se určuje pomocí priority pozic svých hráčů, soupeřových robotů a pozice míče. Pro experimenty v této práci budeme používat výchozí nastavení hlavičky.

Hlavička se, stejně jako všechna pravidla, ukládá do souboru se strategií. Je umístěna na začátku souboru.



Obrázek 4: Strategy Creator



Obrázek 5: Strategy Creator - hlavička strategie

4 Strategie

V této kapitole se již nebudeme zabývat obecným popisem strategií v simulátoru. To je uvedeno v kapitole č. 3 o simulátoru samotném. Popíšeme stávající řešení výběru pravidla ze strategie pro danou situaci a dále se budeme detailněji zajímat o speciální typy pravidel ve strategii. Speciální typy pravidel mohou být pro tým užitečné, ale někdy také na škodu. Pokud jsou daná pravidla správně detekována, může mít jejich odstranění nebo upravení ve strategii důležitý význam.

4.1 Stávající řešení

V průběhu hry je zřejmé, že je třeba určit které pravidlo pro danou herní situaci zvolíme. V simulátoru je implementován algoritmus, kterým je výběr pravidla realizován. Na základě gridových souřadnic hry a daného pravidla se vybere takové pravidlo, které nejlépe popisuje aktuální situaci na herní ploše. V průběhu tohoto výběru se pomocí koeficientů (vah) zohledňuje i typ pravidla - jestli je ofenzivní nebo defenzivní. Stejně tak jsou při výběru zohledněny informace z hlavičky strategie - priority pozic hráčů a míče.

Strategie jsou definovány na základě gridových souřadnic také právě proto, aby byl algoritmus pro výběr pravidla ze strategie jednodušší a nezatěžoval průběh hry. Tím, že se hra dynamicky mění, je třeba zachovat výběr pravidel dostatečně rychlý. Gridový souřadnicový systém zjednodušuje lokalizaci robotů na hřišti. Strategii můžeme zpřesnit, pokud zvýšíme hustotu gridů. Je třeba ale brát v potaz, že čím více gridů na hřišti je, tím více je možných kombinací souřadnic a tím více se může algoritmus pro výběr pravidla zpomalovat.

Mimo hlavní strategii týmu jsou v simulátoru implementovány taktiky. Je tady několik typů taktik - defenzivní, ofenzivní a další. Podrobněji jsou popsány v bakalářské práci Sebastiana Balona [6]. V průběhu hry se někteří hráči v některých podmínkách vymykají hlavní strategii týmu a chovají se podle těchto taktik.

Žádné detekce speciálních pravidel, ani analýzy strategií zatím implementovány nejsou.

4.2 Teoretický základ detekce speciálních pravidel ve strategii

Implementace všech optimalizací strategií vznikala na základě myšlenek uvedených v [2], [3] a [4]. Všechny tyto návrhy bylo potřeba do simulátoru naimplementovat a následně provést dostatečný počet experimentů, které by buď vyvrátily nebo potvrdily jejich platnost. Následně se rozhodlo o dalším využití těchto optimalizací a existenci speciálních pravidel ve strategii.

4.2.1 Cyklická pravidla

Při analýze strategie můžeme nalézt pravidla, která mezi sebou mohou mít speciální vztahy. V našem případě to jsou tzv. *cyklická pravidla*. Několik pravidel ve strategii může vytvořit nekonečný cyklus pohybů hráčů po hřišti, a tím by nikdy nedošlo k útoku a

úspěšnému gólu týmu. Je zřejmé, že tato situace není pro náš tým žádoucí. Proto je dobré takovéto cykly ve strategii detekovat.

Cyklus ve strategii eliminujeme tím, že jedno z pravidel buď odstraníme ze strategie úplně nebo ho jen upravíme tak, aby již k zacyklení nedocházelo. Po každé úpravě strategie je vhodné provést další detekci na cyklická pravidla, jelikož po úpravě se mohou vytvořit cykly s jinými pravidly.

Co je myšleno cyklem ve strategii, čili cyklickými pravidly budeme demonstrovat na následujícím příkladě. Nutno zmínit, že v této optimalizaci se nebere ohled na pozici soupeře a detekce probíhá jen na základě pohybu hráčů našeho týmu. V tomto příkladu zanedbáme i hlavičku strategie a její dodatečné parametry. Proto pro ukázkovou strategii ve výpisu 3 využijeme jen souřadnice robotů našeho týmu.

```
.Rule 1 o Prvni
.Mine 1,2 1,3 2,2 2,3
.Move 2,1 2,4 3,1 3,4

.Rule 2 o Druhe
.Mine 2,1 2,4 3,1 3,4
.Move 3,1 3,4 4,1 4,4

.Rule 3 o Treti
.Mine 3,1 3,4 4,1 4,4
.Move 4,2 4,3 5,2 5,3

.Rule 4 o Ctvrte
.Mine 4,2 4,3 5,2 5,3
.Move 1,2 1,3 2,2 2,3
```

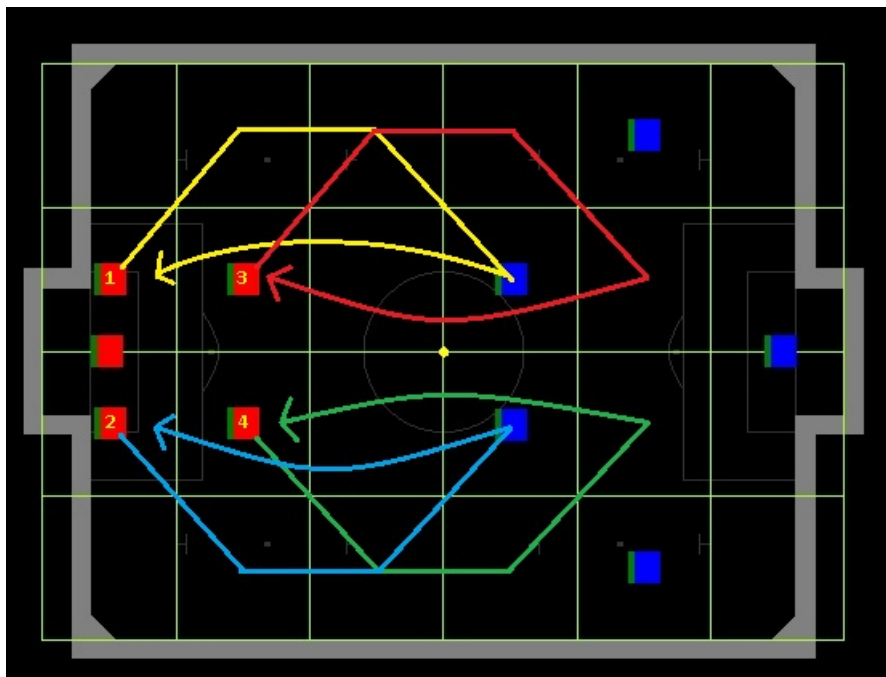
Výpis 3: Výpis strategie s cyklickými pravidly

Ve výpisu vidíme gridové souřadnice 4 robotů - hráčů. Brankář je pátý hráč, který je stále v brance. Pro lepší názornost jsou výše uvedená pravidla z výpisu 3 graficky znázorněna na obrázku č. 6. Roboti na obrázku jsou označeni čísly 1 až 4, což odpovídá gridovým souřadnicím, jak jdou postupně zleva doprava ve výpisu. Každá křivka v jedné barvě zobrazuje pohyb jednoho robota, jak je dán strategií. Z obrázku je zřejmé, že pohyb je cyklický a všichni roboti se po vykonání určitého pohybu vrací do výchozích pozic.

Detekce cyklických pravidel probíhá na základě znalosti poznatků z lineární algebry a teorie grafů. Z každé strategie jsme schopni vytvořit tzv. *matici sousednosti pravidel* R , která je de facto maticí sousednosti orientovaného grafu používanou v teorii grafů. Definujme matici R pomocí definice 4.1 převzaté z [2].

Definice 4.1 Mějme systém dat $A = (U, C \cup D)$, ve kterém $U = \{x_1, \dots, x_n\}$ je množina pravidel ve strategii a $n = |U|$. Indexem značíme číslo pravidla. C je množina gridových souřadnic všech pravidel za klíčovým slovem .Mine (výchozí pozice robotů) a D je množina gridových souřadnic všech pravidel za klíčovým slovem .Move (cílové pozice). Matice sousednosti pravidel, značíme R , je matice $n \times n$, kde

$$r_{ij} = \begin{cases} 1, & \text{pokud } D(x_i) = C(x_j); \\ 0, & \text{v ostatních případech.} \end{cases}$$



Obrázek 6: Cyklický pohyb hráčů na hřišti

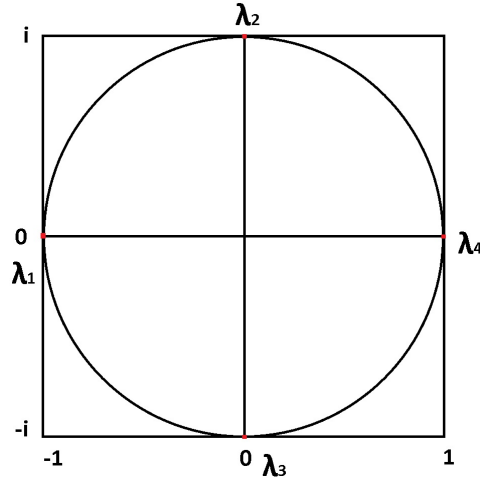
Matice R pro naši ukázkovou strategii je matice (1).

Tato matice je vždy čtvercová, protože počet řádků = počet sloupců = počet pravidel ve strategii. Pokud cílová pozice robotů (souřadnice *.Move*) pravidla č. 1 je stejná jako výchozí pozice robotů (souřadnice *.Mine*) jiného pravidla, v našem případě pravidla č. 2, pak klademe do matice hodnotu 1. Hodnota 1 se klade na pozici: řádek = 1 = pravidlo č. 1 a sloupec = 2 = pravidlo č. 2. Takto sestavíme následující matici R (1).

$$R = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix} \quad (1)$$

V následujícím kroku vypočítáme vlastní čísla matice R (1). Vlastní čísla naší ukázkové matice (1) jsou $\lambda_1 = -1, \lambda_2 = i, \lambda_3 = -i, \lambda_4 = 1$. Tato vlastní čísla mohou být znázorněna v komplexních rovině jako čtyři body, viz obrázek č. 7. Je zřejmé, že tyto čtyři body rozdělují jednotkovou kružnici na čtyři stejné části, což znamená, že tato pravidla tvoří cyklus.

Podle [2] matice, která má nenulová vlastní čísla obsahuje nějaký cyklus pravidel. Z výše uvedeného je zřejmé, že naše strategie cyklus obsahuje. Pokud by matice R měla vlastní čísla nulová, pak můžeme detekci cyklických pravidel ukončit, protože strategie žádný cyklus pravidel neobsahuje.



Obrázek 7: Znázornění vlastních čísel v komplexní rovině

Nyní sestavíme novou matici $S = R^T + R$, kde matice R^T je transponovaná matice R . Matice S pro vzorovou strategii je matice (2).

$$S = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \quad (2)$$

Zmíníme několik důležitých vět a důkazů týkajících se výsledné matice S převzatých z [2].

Věta 4.1 *Nechť $i \in [1, n]$, pokud $s_{ii} = 2$, pak pravidlo x_i je smyčka.*

Důkaz. Podle definice matice S dostáváme

$$s_{ii} = r_{ii} + r_{ii} = 2r_{ii}.$$

Jestliže $s_{ii} = 2$, pak $r_{ii} = 1$, což znamená, že $D(x_i) = C(x_i)$, tj. souřadnice výchozí pozice robotů a cílové souřadnice jsou shodné. Proto je x_i smyčka. ■

Věta 4.2 *Nechť $i, j \in [1, n], i \neq j$, pokud $s_{ij} = 2$, pak $r_{ij} = r_{ji} = 1$, tj. pravidlo x_i a x_j tvoří cyklus.*

Důkaz. Podle definice matice S dostáváme

$$s_{ij} = r_{ij} + r_{ji}.$$

Hodnota r_{ij} je buď 1 nebo 0. Pokud $s_{ij} = 2$, pak $r_{ij} = 1$ a $r_{ji} = 1$. Tudíž souřadnice $D(x_i) = C(x_j)$ a zároveň $D(x_j) = C(x_i)$. Z toho plyne, že x_i a x_j tvoří cyklus. ■

Věta 4.3 *Nechť $j, k \in [1, n]$, $j + k \leq n$, pro $i = j, \dots, j + k$ platí, že pokud $\exists s_{iu} = 1$ a $\exists s_{iv} = 1$, $u, v \in [j, j + k]$, $u \neq v$ pak x_i je členem cyklu.*

Důkaz věty 4.3 zde rozebírat nebudeme. Z výše uvedených vět je zřejmé, jakým způsobem lze detekovat cykly ve strategii.

Ve vzorové strategii z výpisu 3 se nevyskytuje žádná smyčka. Cyklus tvoří všechna pravidla následovně $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 1$.

4.2.2 Hub pravidla

Každou strategii lze vyjádřit orientovaným grafem, kde vrcholy grafu jsou pravidla a orientované hrany jsou směry, do kterých dalších pravidel bude hra přecházet, čili kam se naši roboti budou přesouvat. Hub pravidla jsou speciálním typem pravidel ve strategii. Jsou to pravidla, která tvoří jakousi křižovatku, úzké místo v orientovaném grafu strategie.

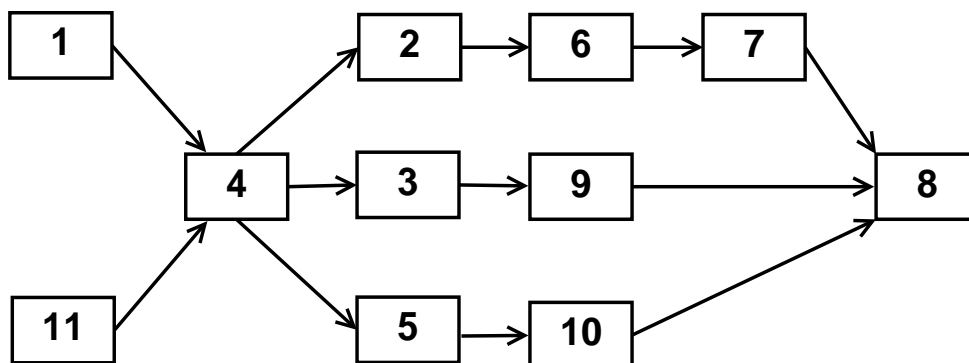
Pokud by se útočníkovi podařilo hru zablokovat tak, že se náš tým do hub pravidla nebude moci dostat, tak to znamená, že i do všech dalších pravidel za hub pravidlem se nebudeme moci dostat. Tato situace může mít fatální důsledky na výsledek hry. Proto je potřeba taková pravidla objevit a následně je buď ze strategie odstranit nebo upravit strategii tak, aby se tam hub pravidla nevyskytovala.

Opět si problematiku hub pravidel ukážeme na vzorové strategii, tentokrát z výpisu 4. Stejně jako v kapitole 4.2.1 o cyklických pravidlech, zanedbáme gridové souřadnice soupeře a hlavičku strategie.

.Rule 1 o První	.Rule 7 o Sedmé
.Mine 2,2 2,3 1,2 1,3	.Mine 3,3 3,4 2,3 3,4
.Move 3,1 3,4 2,1 2,4	.Move 4,4 3,4 3,4 5,3
.Rule 2 o Druhé	.Rule 8 o Osme
.Mine 3,2 4,4 2,2 2,4	.Mine 4,4 3,4 3,4 5,3
.Move 3,3 3,4 2,2 5,4	.Move 4,4 3,4 6,3 6,3
.Rule 3 o Třetí	.Rule 9 o Deváté
.Mine 3,2 4,4 2,2 2,4	.Mine 2,2 4,4 1,2 2,4
.Move 2,2 4,4 1,2 2,4	.Move 4,4 3,4 3,4 5,3
.Rule 4 o Čtvrté	.Rule 10 o Desáté
.Mine 3,1 3,4 2,1 2,4	.Mine 4,4 4,4 4,4 5,4
.Move 3,2 4,4 2,2 2,4	.Move 4,4 3,4 3,4 5,3
.Rule 5 o Páté	.Rule 11 o Jedenácté
.Mine 3,2 4,4 2,2 2,4	.Mine 1,1 2,2 3,4 5,4
.Move 4,4 4,4 4,4 5,4	.Move 3,1 3,4 2,1 2,4
.Rule 6 o Šesté	
.Mine 3,3 3,4 2,2 5,4	
.Move 3,3 3,4 2,3 3,4	

Výpis 4: Ukázková strategie s hub pravidly

Orientovaný graf vzorové strategie je na obrázku č. 8.



Obrázek 8: Orientovaný graf strategie s huby

Abychom mohli hub pravidlo detekovat, budeme muset pravidla ve strategii nějakým způsobem ohodnotit podle jejich důležitosti. Toto ohodnocení plyne z počtu vstupních a výstupních hran vrcholu grafu = pravidla. Pravidlo se stává silnějším, pokud z něj vede více výstupních hran, jelikož ovlivňuje ve strategii více ostatních pravidel a tudíž má i větší vliv na průběh hry. Nyní bychom scóre pravidla počítali jako součet scóre pravidel, do kterých z něj vede výstupní hrana. Scóre pravidla označíme x s indexem čísla pravidla. Ve vzorové strategii by například výpočet scóre pro pravidlo č. 4 vypadalo následovně $x_4 = x_2 + x_3 + x_5$.

Je třeba zmínit, že scóre nepočítáme u pravidel, která vedou sami do sebe, tzv. *smyček*. Tato pravidla mají stejné gridové souřadnice *.Mine* a *.Move*. Smyčky v orientovaném grafu tvoří slepou cestu a proto je ve strategii nepotřebujeme (z pohledu analýzy hub pravidel) a můžeme je ze strategie odstranit. Pokud je ve strategii necháme, pak je třeba mít na paměti, že na smyčky nebereme ohled při výpočtu scóre jednotlivých pravidel.

Nicméně nelze říct, že pravidla, do kterých výstupní hrany vedou jsou rovnocenná. Některá jsou méně, některá více důležitá a proto nám nestačí výpočet scóre jen podle výstupních hran. Podle [4] musíme vzít v úvahu také vstupní hrany do pravidla, jejichž počet snižuje jeho důležitost ve strategii. Scóre pravidla spočítáme jako součet podílů scóre pravidel, do kterých vede výstupní hrana a počtu vstupních hran do těchto pravidel.

Pro vyčíslení scóre pravidla mějme sadu n pravidel. Nechť $L_k \subset \{1, 2, \dots, n\}$ je množina pravidel, která mají vstupní hranu z pravidla k , tj. L_k je množina výstupních hran pravidla k . Pro každé pravidlo k potřebujeme vyčíslit jeho scóre dle následující rovnice

$$x_k = \sum_{j \in L_k} \frac{x_j}{n_j} \quad (3)$$

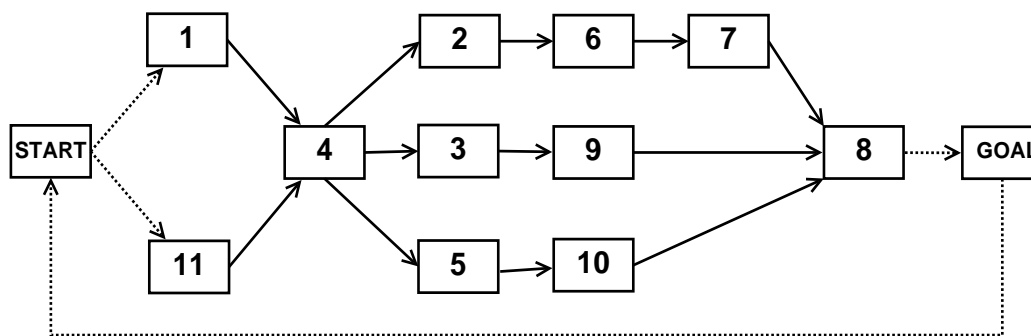
kde n_j je počet vstupních hran do pravidla j .

Pro vzorovou strategii uvádíme výpočty pro všechna pravidla:

$$\begin{aligned}
 x_1 &= \frac{x_4}{2} & x_7 &= \frac{x_8}{3} \\
 x_2 &= \frac{x_6}{1} & x_8 &= \frac{GOAL}{1} \\
 x_3 &= \frac{x_9}{1} & x_9 &= \frac{x_8}{3} \\
 x_4 &= \frac{x_2}{1} + \frac{x_3}{1} + \frac{x_5}{1} & x_{10} &= \frac{x_8}{3} \\
 x_5 &= \frac{x_{10}}{1} & x_{11} &= \frac{x_4}{2} \\
 x_6 &= \frac{x_7}{1}
 \end{aligned}$$

Do strategie v našem postupu jsou přidána dvě jakási pomyslná pravidla, aby všechna pravidla ve strategii měla alespoň jednu vstupní a jednu výstupní hranu. Tím při výpočtu nedojde k nedefinovanému dělení nulou, ani k výsledku skóre 0, což by nedávalo smysl, protože každé pravidlo ve strategii může mít nějaký, byť třeba minimální význam a vliv na hru. Přidaná pomocná pravidla jsme pojmenovali *START* a *GOAL*. *START* je pravidlo, ze kterého vedou všechny výstupní hrany do prvních pravidel, tzn. do pravidel jejichž gridové souřadnice *Mine* se neshodují s žádnými gridovými souřadnicemi *Move* jiného pravidla ze strategie.

GOAL je pravidlo, do kterého vedou všechny vstupní hrany z posledních pravidel strategie, tzn. z pravidel, jejichž gridové souřadnice *Move* nejsou shodné s žádnými gridovými souřadnicemi *Mine* jiného pravidla ve strategii. Po této úpravě by naše vzorová strategie vypadala jako na obrázku č. 9.



Obrázek 9: Orientovaný graf strategie s přidáním pravidel *GOAL* a *START*

Podle výše popsaného postupu jsme schopni pro každé pravidlo napsat rovnici, podle které bychom vyjádřili skóre pravidla v dané strategii. Sestavíme takové rovnice pro všechna pravidla a získanou soustavu lineárních rovnic přepíšeme do maticového tvaru. Maticový zápis pro vzorovou strategii můžeme vidět v matici (4). Jsou v ní zařazeny také pomocná pravidla *GOAL* a *START*, jejichž skóre je označeno jako *G* a *S*.

$$\begin{pmatrix}
0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0
\end{pmatrix}
\begin{pmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6 \\
x_7 \\
x_8 \\
x_9 \\
x_{10} \\
x_{11} \\
G \\
S
\end{pmatrix}
=
\begin{pmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6 \\
x_7 \\
x_8 \\
x_9 \\
x_{10} \\
x_{11} \\
G \\
S
\end{pmatrix} \quad (4)$$

Po krátkém prozkoumání matice si uvědomíme, že tato soustava lineárních rovnic je homogenní. Odečtením vektoru pravých stran se nám hodnoty na diagonále sníží o 1 a na pravé straně budou ve všech řádcích nuly. U homogenních soustav lineárních rovnic buď existuje jen triviální řešení, tzn. že všechna $x = 0$ nebo nekonečně mnoho řešení vyjádřeno parametrem⁴. Při detekci hub pravidel nás bude zajímat řešení parametrické, podle kterého určíme skóre pravidel ve strategii. Ovšem mohou existovat strategie, u kterých nebude možno hub pravidla detekovat právě z důvodu neexistence parametrického řešení homogenní soustavy lineárních rovnic.

Po vyřešení soustavy (4) dostáváme skóre pro pravidla $x_1 = 9,5; x_2 = 6,3; x_3 = 6,3; x_4 = 19; x_5 = 6,3; x_6 = 6,3; x_7 = 6,3; x_8 = 19; x_9 = 6,3; x_{10} = 6,3; x_{11} = 9,5; G = 19$ a $S = 19$. K řešení využíváme jako první parametr hodnotu 100. Všimněme si, že pomocná pravidla *START* a *GOAL* mají také vysoké skóre a je to zřejmé, protože z nich se dostáváme do všech ostatních pravidel.

Otázkou stále zůstává, od které hodnoty skóre považovat pravidlo za hub a při kterých hodnotách to ještě hub není. V našem řešení využijeme aritmetický průměr všech hodnot skóre a tento průměr stanovíme jako hraniční limit. Pravidla, která mají větší skóre než je limit, jsou posuzována jako hub pravidla. Při určování hub pravidel již nebereme v úvahu pomocná pravidla *START* a *GOAL*.

Ve vzorové strategii je limit 10,72 a proto jsme stanovili **pravidla č. 4 a č. 8** jako hub pravidla (mají skóre 19, což je vyšší než limit).

Hub pravidla jsme našli a následuje problém, co s nimi. Někdy je vhodné je ve strategii vůbec nemít, někdy je lepší je mírně upravit tak, aby hubem již nebyla.

Pokud je hubem pravidlo, které následuje ihned za pomocným pravidlem *START* nebo bezprostředně před pomocným pravidlem *GOAL* můžeme jej ze strategie odstranit a nenahrazovat ho jinými pravidly. Jakmile pravidlo následuje po pravidlu *START*, pak se při jeho odstranění stane jen to, že následovníci odstraněného pravidla se napojí na *START*, čímž neovlivníme průběh hry a úspěšně odstraníme hub pravidlo ze strategie. Na

⁴Soustavy rovnic a možnosti jejich řešení:
<http://homepages.math.slu.cz/AdamHlavac/Soustavyrovnic.pdf>

stejném principu je postaveno odstranění pravidla, které stojí bezprostředně před pravidlem *GOAL* a je označeno za hub. Jakmile ho odstraníme, pak předchůdci odstraněného pravidla se pomyslně napojí na pravidlo *GOAL*. Toto je případ pravidla č. 8 v ukázkové strategii. Tudíž ho můžeme bez obav smazat a nenahrazovat.

Jiným způsobem se budeme chovat ke strategii, pokud hub pravidlo nalezneme někde „uprostřed“ grafu pravidel. Pokud bychom pravidlo jen odstranili, pak bychom se hubu nezbavili. Pravidlo by sice ze strategie zmizelo, ale při další analýze strategie bychom zjistili, že se hub pravidlem stala pravdila nebo pravidlo, které předcházelo původnímu hubu. Proto naším cílem bude hub pravidlo mírně upravit a přidat další pravidla, která nahradí hub (úzké místo) několika různými nezávislými větvemi v grafu strategie. Počet větví je dán počtem předchůdců hub pravidla. Pokud je počet následujících větví větší než počet předchůdců, pak se vytvoří jen tolik nových pravidel, kolik je předchůdců. Zbýlá následující pravidla se po odstranění hubu připojí na pravidlo *START*.

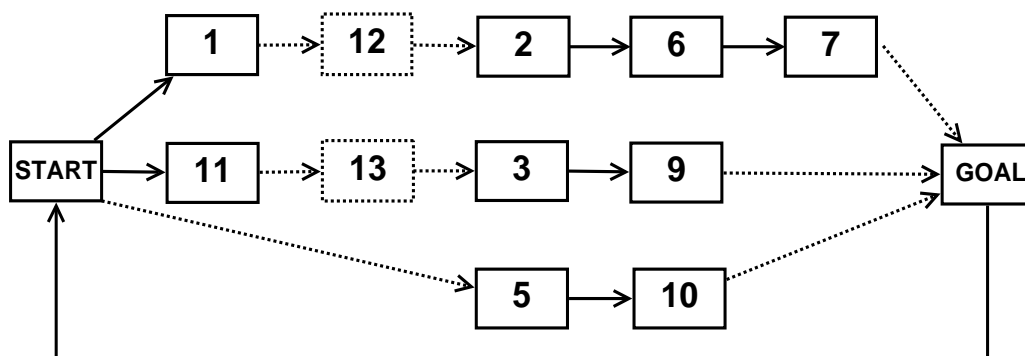
Abychom korektně nahradili hub pravidlo a jiné už nevzniklo, tak je třeba si uvědomit, že souřadnice *.Mine* následovníků hubu se musí lišit, proto je třeba je upravit. Budeme přidávat nové pravidlo místo hubu, označme ho *newrule*. Gridové souřadnice robotů *.Mine* i *.Move* pravidla *newrule* musíme vypočítat. Mohli bychom je určit náhodně bez jakéhokoli zamyšlení, ale pravděpodobně bychom neudrželi význam a smysl návaznosti pravidel ve strategii. Proto tyto souřadnice spočítáme podle vzdáleností pozic jednotlivých robotů na hřišti. Známe cílovou pozici (*.Move*) následovníka hubu a výchozí pozici (*.Mine*) předchůdce hubu. Pro každou větev zvlášť, pro každé nově přidané pravidlo, vypočítáme vzdálenost mezi souřadnicemi *.Mine* předchůdce a souřadnicemi *.Move* následníka a do její poloviny položíme nové pravidlo *newrule*. Následně podle výpočtu musíme upravit souřadnice *.Move* předchůdce a *.Mine* následovníka, aby na *newrule* vznikly vazby.

Cílem výpočtů a změn souřadnic pravidel okolo hubu je rozdělení větví. Po správné analýze a nahrazení všech hub pravidel ve vzorové strategii z výpisu č. 4 dojdeme ke strategii vyjádřené orientovaným grafem na obrázku č. 10. Tečkovanou čarou jsou značená nově vytvořená pravidla a nové vazby, které se musely oproti původní strategii změnit a upravit. Z obrázku je zřejmé, že úplně zmizelo původní pravidlo č. 8 a pravidlo č. 4 bylo nahrazeno novými pravidly č. 12 a 13.

4.2.3 Útočná pravidla

Strategie mohou být útočné nebo obranné, stejně jako tomu je v lidském fotbale. V klasické hře je docela jednoduché rozeznat situaci, kdy je tým v útoku a kdy v obraně. V robotické fotbale tato situace není na první pohled úplně zřejmá. Pro zjednodušení budeme považovat za útočný tým ten, který aktuálně drží míč. Dobře známé pravidlo říká, že nejlepší obrana je útok a proto se zaměříme na generování útočných pravidel a tím možné vylepšení strategie.

Každá hra odehraná v simulátoru je logována do souboru, kde se ukládají jak reálné, tak gridové souřadnice pozic všech hráčů a míče na hřišti v každém tiku hry. Generování útočných pravidel je založeno na docela primitivním učení strategie týmu během odehra-



Obrázek 10: Graf strategie z výpisu č. 4 po nahrazení hubů odpovídající výpisu č. 5

ných her. Je to dynamické zlepšování týmu na základě zkušeností v odehraných zápasech a obdržení gólů.

Je dobré si uvědomit, že síla strategie není v kvantitě pravidel, ale v jejich kvalitě. Mnohdy strategie se 40 pravidly prohraje oproti jiné, lépe sestavené, strategii pouze s 10 pravidly. Naším cílem bude strategii obohatit o pravidla, která pro ni mohou být významná tak, aby tým byl úspěšnější. Úspěch týmu může být způsobem také kvalitní obranou. Tato práce se ovšem vylepšením defenzívy nezabývala.

Generování ofenzivních pravidel je podle [3] založeno na docela jednoduché myšlence. V průběhu hry se po každém gólu hra pozastaví a my můžeme nahlédnout do logovacího souboru a naučit se něco z dosavadního chování soupeře. Této myšlenky využijeme v případě, kdy dostaneme gól od soupeře. Během přerušení hry se podíváme do logu a pravidlo, které bezprostředně předcházelo gólové situaci se naučíme, čili přidáme do naší strategie.

Po krátkém zamyšlení je zřejmé, že bude hodně záležet na ostatních pravidlech strategie, zda se přidání nového pravidla projeví nebo ne. Není totiž jisté, že k přidanému „gólovému“ pravidlu vůbec dojde. Pokud ostatní pravidla strategie na něj nenavazují, tak během hry nebude vybráno a tým bude hrát stejně, jakoby k žádnému učení nedošlo.

5 Praktická implementace detekce speciálních pravidel

V následující kapitole si popíšeme praktickou část této práce. Všechny navržené optimalizace (viz kapitola 4) a vylepšení strategií bylo třeba naimplementovat do simulátoru, který vznikl a stále se vyvíjí na katedře informatiky. Simulátor před implementací obsahoval `StrategyCreator`, ve kterém se strategie mohou tvořit, ale neobsahoval žádnou analýzu ani optimalizaci těchto strategií.

Do `StrategyLibrary` jsme přidali novou třídu s názvem `SG.StrategyOptimization`, která obsahuje téměř celou implementaci všech tří optimalizačních algoritmů, které jsme v práci popsali.

Třída `SG.StrategyOptimization` je rozdělena na tři regiony, ve kterých jsou metody, které se využívají pro optimalizace a zlepšení popsaná v kapitole 4.2. Byla přidána také třída `SG.MatrixSolver`, ve které jsou metody pro výpočty a operace s maticemi.

Již v konstruktoru třídy `SG.StrategyOptimization` se počítá matice sousednosti pravidel, jelikož se hojně využívá pro analýzu jak hub pravidel, tak pro nalezení cyklických pravidel. Bylo potřeba upravit i kód třídy `RS.Simulator`, `RSS.FormStrategyCreator` a `RSS.FormSimulator`.

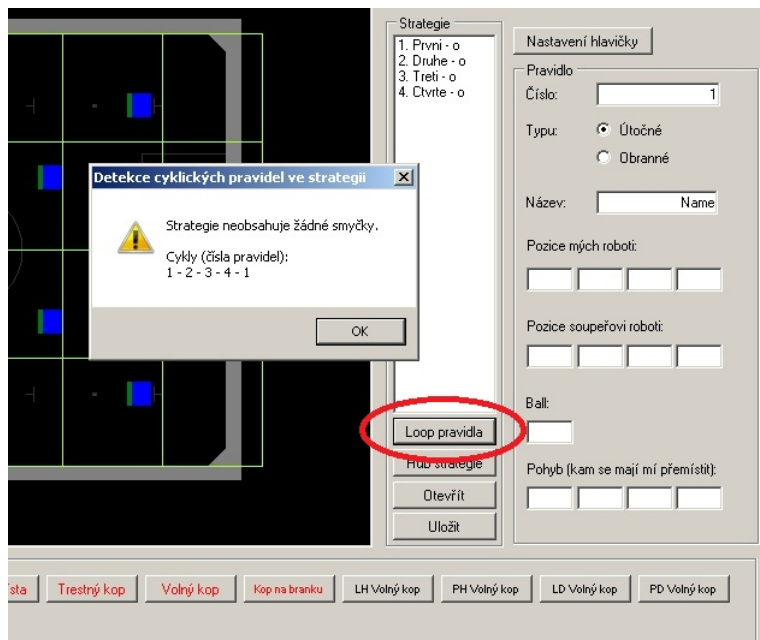
5.1 Cyklická pravidla

V kódu týkajícího se nalezení cyklických pravidel jsou následující metody, které stručně popíšeme.

- `CountFinalMatrixS` - ze zdrojové matice sousednosti pro danou strategii vypočítá matici S , ze které se následně detekují cykly; vrací výslednou matici S .
- `DetectionSelfLoopRules` - prohledá matici S a najde v ní smyčky, vrací seznam čísel pravidel, která jsou smyčky.
- `DetectSelfLoopRules` - prohledá matici S a najde v ní smyčky, vrací seznam indexů v matici S , kde byly nalezeny smyčky.
- `ExistsLoopRules` - ověřuje existenci cyklických pravidel v dané strategii pomocí metod ve třídě `SG.MatrixSolver`, které vypočítají vlastní čísla pro matici R (původní matice sousednosti pravidel) a podle výsledku rozhoduje, zda se ve strategii vyskytují nebo nevyskytují cyklická pravidla.
- `DetectionLoopRules` - prohledá matici S a hledá v ní cyklická pravidla. Vzhledem k tomu, že strategie může obsahovat i více cyklů, tak metoda vrací seznam polí s čísly pravidel tak, jak jsou za sebou v cyklu.
- `CheckLoop` - pro daný cyklus kontroluje v původní matici R , zda jde opravdu o cyklus a zda má poslední pravidlo cyklu vazbu na první pravidlo.

V implementaci bylo přidáno tlačítko do `RSS.FormStrategyCreator\verb`, aby po načtení dané strategie ze souboru mohla být strategie analyzována na cyklická pravidla.

Jak simulátor detekoval cyklická pravidla pro naši ukázkovou strategii z výpisu č. 3 můžeme vidět na obrázku č. 11.



Obrázek 11: Screenshot implementace analýzy cyklických pravidel v simulátoru

5.2 Hub pravidla

V implementaci analýzy hub pravidel jsme do kódu zařadili automatické odstranění detekovaných pravidel, podle postupu popsaného v kapitole 4.2.2. Po odstranění hub pravidel se vytvoří nový soubor se strategií s názvem *puvodniNazev_NEW.strg*. Po analýze se uživateli samozřejmě zobrazí dialogové okno, které ho informuje o výsledcích analýzy hub pravidel, score jednotlivých pravidel a odstranění hubů. Region kódu týkajícího se detekce hub pravidel obsahuje metod více. Zmíníme jen ty nejdůležitější.

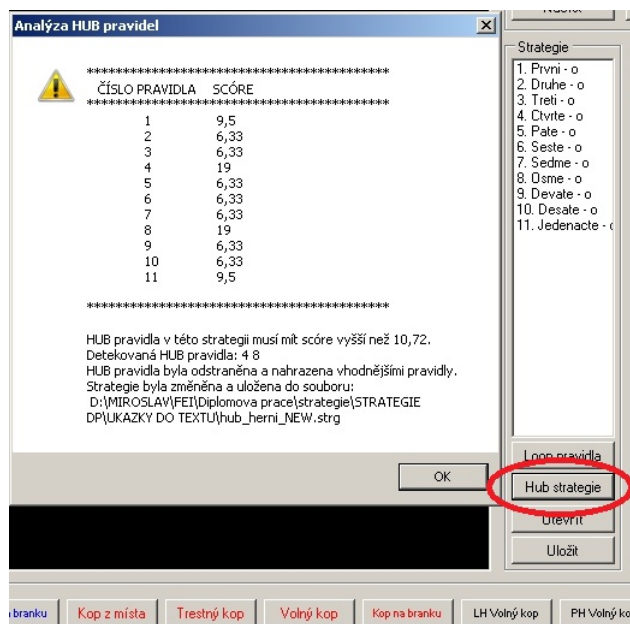
- `FindHubRules` - shrnující metoda, hledá hub pravidla ve strategii, vrací seznam čísel pravidel, která jsou detekována jako hub.
- `CountCoefficientMatrix` - nad zadanou maticí sousednosti (matice R) spočítá matici koeficientů podle příchozích a odchozích hran z resp. do pravidel; následně sníží na hlavní diagonále prvky o 1, čímž odečte vektor pravých stran a vrací homogenní matici s koeficienty pro výpočet score pravidel.
- `CountIncidenceMatrix` - spočítá matici sousednosti pravidel v dané strategii tak, že umístí hodnotu 1 na pozici x, y , pokud se z pravidla na řádku x dostaneme do pravidla ve sloupci y . Zbývající hodnoty jsou 0.

- `CountSpecialIncidenceMatrix` - spočítá matici sousednosti pravidel ve strategii stejně jako `CountIncidenceMatrix`, ale navíc přidá do matice pomocná pravidla *START* a *GOAL*. Pravidlo *START* vyjadřuje poslední řádek matice a pravidlo *GOAL* předposlední řádek matice. V této matici jsou vynechána pravidla (matice je neobsahuje), která byla označena jako *smyčky*.
- `CompareRules` - porovnává identitu dvou pravidel, resp. gridových souřadnic daných pravidel. Vstupem je pole třídy `Vector2D`, což jsou např. souřadnice *.Mine* nebo *.Move*. Pokud obsahují stejné souřadnice, ikdyž v jiném pořadí, pak metoda vrací hodnotu `true`. Jinak `false`.
- `CountReplacedRule` - na základě vstupu = pravidla následník a předchůdce hubu, metoda spočítá nové pravidlo, které by mělo být doplněno do strategie místo detekovaného hub pravidla. Nově vytvořené pravidlo má název *Replaced-HUB* a pod tímto názvem ho nalezneme i v upravené strategii. Tato metoda také počítá upravené souřadnice pro předchůdce a následníka hub pravidla podle postupu popsáném v kapitole 4.2.2. Metoda také mění název následovníků na *puvodniNazev-CHANGED*.
- `ReplaceHub` - vstupem je číslo hub pravidla, které bylo detekováno a má být odstraněno nebo upraveno. Podle postupu popsáném v kapitole 4.2.2 metoda odstraní bez náhrady hub pravidlo, jehož předchůdce je pravidlo *START* nebo následník *GOAL*. Jiná pravidla nahradí novými vhodnými pravidly pro každého předchůdce hub pravidla. Souřadnice předchůdců hub pravidla musí být upraveny a proto je upraven i název pravidla na *puvodniNazev-CHANGED*.

Do `StrategyCreatoru` bylo přidáno tlačítko pro detekci hubů. Hub pravidla je třeba najít a upravit před zápasem, případně při přerušení zápasu. Hub analýza musí probíhat vždy nad kompletní strategií, aby měla význam a byla objevena všechna hub pravidla.

Na obrázku č. 12 můžeme vidět výsledek analýzy hub pravidel ze vzorové strategie z výpisu 4. Je zřejmé, že teoretický výpočet i praktická implementace si odpovídají a simulátor také detekoval hub pravidla č. 4 a 8. Při detekci hub pravidel simulátor automaticky hub pravidla odstraní a případně nahradí jinými vhodnými pravidly. Uživatel je následně informován, do kterého souboru se nová upravená strategie zapsala.

Ve výpise 5 můžeme vidět upravenou původní strategii z výpisu 4 po hub detekci. V nové strategii můžeme vidět pravidla s upravenými souřadnicemi. Strategie již vůbec neobsahuje hub pravidla č. 4 a 8, ale byla obohacena o pravidla č. 12 a 13 oproti původní strategii. Strategie ve výpise 5 odpovídá grafu na obrázku č. 10.



Obrázek 12: Screenshot implementace analýzy hub pravidel v simulátoru

.Rule 1 o První—CHANGED	.Rule 9 o Devate
.Mine 2,2 2,3 1,2 1,3	.Mine 2,2 4,4 1,2 2,4
.Move 3,2 3,4 1,2 1,4	.Move 4,4 3,4 3,4 5,3
.Rule 2 o Druhé—CHANGED	.Rule 10 o Desate
.Mine 3,3 3,4 1,2 3,4	.Mine 4,4 4,4 4,4 5,4
.Move 3,3 3,4 2,2 5,4	.Move 4,4 3,4 3,4 5,3
.Rule 3 o Třetí —CHANGED	.Rule 11 o Jedenacte—CHANGED
.Mine 2,2 4,4 1,3 3,4	.Mine 1,1 2,2 3,4 5,4
.Move 2,2 4,4 1,2 2,4	.Move 2,2 3,3 2,3 4,4
.Rule 5 o Pate	.Rule 12 o Replaced—HUB
.Mine 3,2 4,4 2,2 2,4	.Mine 3,2 3,4 1,2 1,4
.Move 4,4 4,4 4,4 5,4	.Move 3,3 3,4 1,2 3,4
.Rule 6 o Seste	.Rule 13 o Replaced—HUB
.Mine 3,3 3,4 2,2 5,4	.Mine 2,2 3,3 2,3 4,4
.Move 3,3 3,4 2,3 3,4	.Move 2,2 4,4 1,3 3,4
.Rule 7 o Sedme	
.Mine 3,3 3,4 2,3 3,4	
.Move 4,4 3,4 3,4 5,3	

Výpis 5: Upravená strategie po hub detekci

5.3 Útočná pravidla

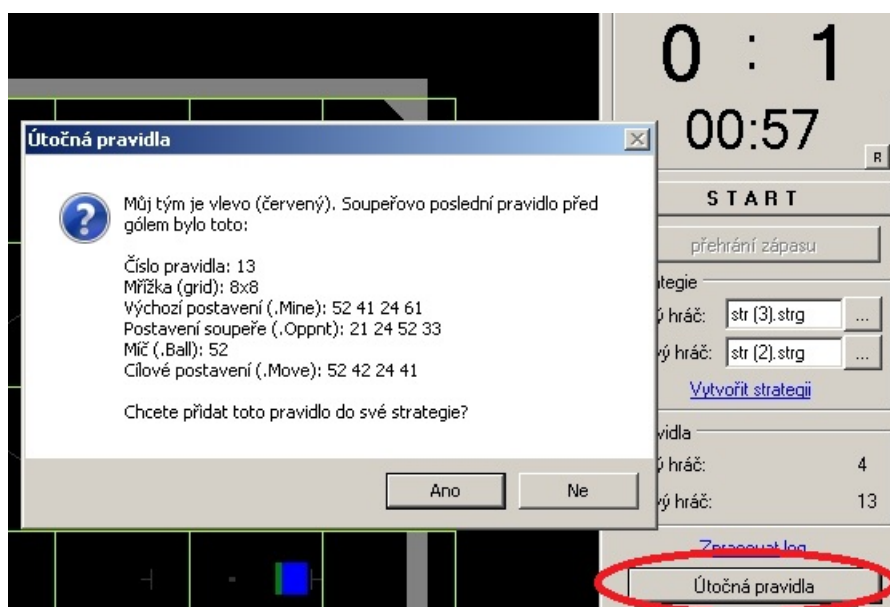
Toto vylepšení strategií probíhá dynamicky během hry, jelikož je založeno na učení útočných soupeřových gólových pravidel. Generování útočných pravidel bylo implementováno přímo do simulátoru, ne do `StrategyCreatoru` jako předchozí optimalizace. Je to právě proto, že ho budeme využívat během hry.

V simulátoru jsme přidali tlačítko „Útočná pravidla“, které se stává aktivním jen v případě, když dostaneme od protihráče gól. Je na uživateli, zda se chce útočná pravidla učit a přidat si je do své strategie nebo ne.

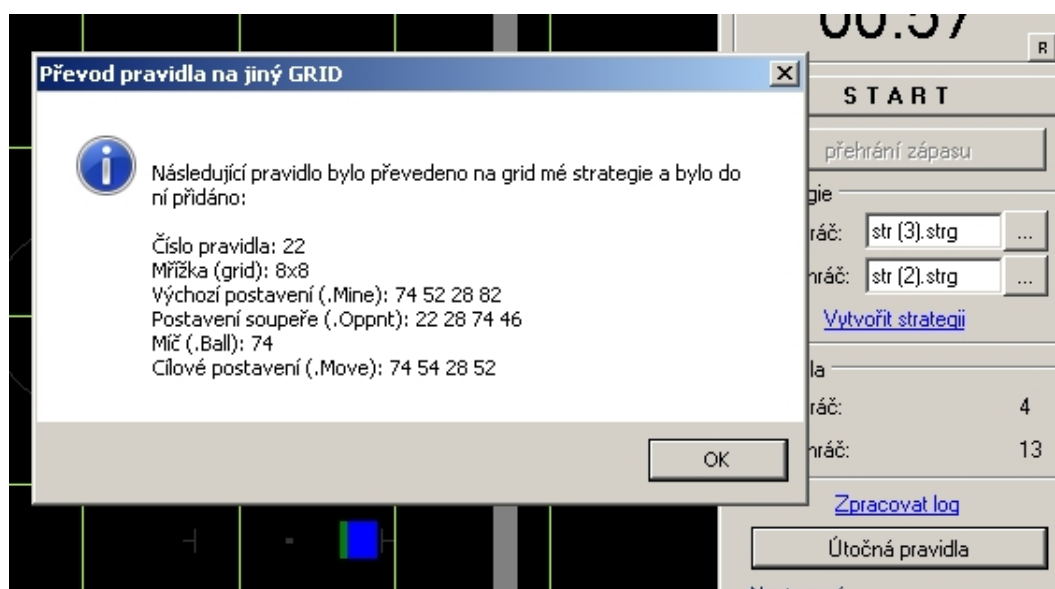
Metody pro generování ofenzivních pravidel byly implementovány opět do třídy `SG.StrategyOptimization` a v následujících bodech si je stručně popíšeme.

- `GetGoalRule` - vrací poslední soupeřovo pravidlo, které použil ve své strategii bezprostředně před tím, než dal gól našemu týmu. Metoda pravidlo získá z logovacího souboru hry.
- `TransformGridRule` - zajišťuje přepočítání souřadnice získaného pravidla v případě, že grid soupeřovy strategie není stejný jako grid našeho týmu. Vrací pravidlo již upravené pro naši strategii.
- `CheckSameRule` - kontroluje, zda se v naší strategii již nevyskytuje pravidlo stejné jako to, které bychom chtěli přidat od soupeře.
- `AddRuleToStrategyFile` - přidává nové pravidlo do naší strategie a zároveň ho zapíše do zdrojového souboru strategie.

Na obrázku č. 13 můžeme vidět dotaz na přidání pravidla do naší strategie po obdržení gólu. Pokud pravidlo přidat chceme, pak proběhne kontrola, zda již naše strategie toto pravidlo neobsahuje a případně nás na to program upozorní a pravidlo nepřidá. Pokud má naše strategie jiný grid než soupeřova, pak se souřadnice přepočítají a opět je na to uživatel upozorněn a pravidlo se mu zobrazí s novými souřadnicemi v gridu jeho strategie. Tato situace je prezentována na obrázku č. 14. Přidané nové pravidlo se řadí vždy na konec strategie.



Obrázek 13: Screenshot implementace generování útočných pravidel v simulátoru



Obrázek 14: Screenshot převodu pravidla na jiný grid

6 Experimenty

V následující kapitole představíme testovací zápasy, jejich výsledky a hodnocení odehraných her při využití uvedených optimalizačních strategií týmů. Následně získané výsledky ohodnotíme a rozmyslíme, zda navržené optimalizační algoritmy strategií budou mít své využití nebo se v praxi neověřily.

Všechny hry mají délku 2 minuty, pokud není uvedeno jinak. Testovací hry byly odehrány na různých počtech strategií od různých autorů s různými počty pravidel. Strategie značíme pro lepší přehlednost pořadovými čísly. Vždy porovnáváme výsledek hry původních strategií, bez jakéhokoliv zásahu našich analýz s výsledkem hry se strategií, která prošla jednou nebo více optimalizacemi. S každou dvojicí strategií je odehrána právě jedna hra, jelikož výsledek hry se stejnými strategiemi se nemění. Toto vychází z vlastností algoritmu. Pokud máme stejné vstupy (strategie) a podmínky hry (fyzika, simulátor, počet hráčů, stejný PC atd.), pak musí být i stejný výstup hry, čili výsledek zápasu.

Experimentální zápasy jsme odehráli nad stejnou sadou strategií pro všechny různé testy. Proto ve všech částech této kapitoly dodržujeme jednotné značení strategií. Ne všechny strategie mohou být využity pro všechny druhy optimalizace. Jak již bylo řečeno, v některých strategiích nelze určit hub pravidlo, v některých zápasech nepadne gól a proto nemůžeme využít generování ofenzivních pravidel apod.

6.1 Analýza cyklických pravidel

V této podkapitole představíme testovací zápasy se strategiemi, které obsahují cyklická pravidla. Celkově jsme odehráli **50 zápasů**. Výsledky zápasů popisuje tabulka č. 1.

V levé části tabulky jsou výsledky zápasů původních strategií bez žádných optimalizací a úprav. V tabulce jsou také uvedeny konkrétní cykly, které jsme pomocí analýzy ve strategii objevili. V pravé polovině tabulky jsou výsledky her po odstranění cyklů ze strategie. Hry jsme odehráli jen se strategiemi, které cyklická pravidla obsahovala, abychom mohli zápasy porovnat s původními výsledky bez optimalizace.

Po analýze cyklů ve strategii potřebujeme cykly ze strategie odstranit, tzn. odstranit nebo změnit některé z pravidel v cyklu tak, aby již na sebe pravidla nenavazovala. My jsme volili jednoduché odstranění vždy jednoho pravidla z jednoho cyklu. V pravé části tabulky č. 1 je vždy uvedeno, která konkrétní pravidla byla ze strategie odstraněna a výsledek zápasu s takovou strategií.

Červeným textem jsou zvýrazněny zápasy, které se oproti původní hře bez optimalizací projeví jako úspěšnější. Úspěšnějším zápasem rozumíme zápas, ve kterém oproti původnímu náš tým vyhrál nebo se snížil počet gólů soupeře.

Základní zápasy bez optimalizace						Zápasy po analýze cyklů		
#	Náš tým	Soupeř	Scóre		Cykly	Scóre		Odstraněné prav.
1	str1	str3	1	1	39-7 11-12-14-15-16	1	1	č. 7 a 11 ze str1
2						0	0	č. 7 a 12 ze str1
3						1	0	č. 7 a 14 ze str1
4						1	1	č. 7 a 15 ze str1
5						1	1	č. 7 a 16 ze str1
6						1	1	č. 39 a 11 ze str1
7						0	0	č. 39 a 12 ze str1
8						1	0	č. 39 a 14 ze str1
9						1	1	č. 39 a 15 ze str1
10						1	1	č. 39 a 16 ze str1
11	str1	str5	0	2	39-7 11-12-14-15-16	0	0	č. 7 a 11 ze str1
12						0	1	č. 7 a 12 ze str1
13						0	0	č. 7 a 14 ze str1
14						0	0	č. 7 a 15 ze str1
15						0	0	č. 7 a 16 ze str1
16						0	2	č. 39 a 11 ze str1
17						0	2	č. 39 a 12 ze str1
18						0	2	č. 39 a 14 ze str1
19						0	2	č. 39 a 15 ze str1
20						0	2	č. 39 a 16 ze str1
21	str5	str3	1	1	25-26	1	1	č. 25 ze str5
22						1	1	č. 26 ze str5
23	str5	str4	0	0	25-26	0	0	č. 25 ze str5
24						0	0	č. 26 ze str5
25	str5	str6	0	1	25-26	0	1	č. 25 ze str5
26						0	1	č. 26 ze str5
27	str5	str7	0	1	25-26	0	1	č. 25 ze str5
28						0	1	č. 26 ze str5
29	str5	str16	0	1	25-26	0	1	č. 25 ze str5
30						0	1	č. 26 ze str5
31	str15	str1	0	0	8-10	0	0	č. 9 ze str15
32						1	1	č. 10 ze str15
33	str15	str2	0	0	8-10	1	0	č. 8 ze str15
34						0	0	č. 10 ze str15
35	str15	str3	0	0	8-10	1	0	č. 8 ze str15
36						0	0	č. 10 ze str15
37	str15	str4	0	0	8-10	0	0	č. 8 ze str15
38						0	0	č. 10 ze str15
39	str15	str16	1	2	8-10	0	0	č. 8 ze str15
40						1	2	č. 10 ze str15
41	str1	str15	0	1	39-7 11-12-14-15-16	0	1	č. 7 a 11 ze str1
42						0	0	č. 7 a 12 ze str1
43						0	0	č. 7 a 14 ze str1
44						2	1	č. 7 a 15 ze str1
45						2	0	č. 7 a 16 ze str1
46						0	2	č. 39 a 11 ze str1
47						0	0	č. 39 a 12 ze str1
48						0	0	č. 39 a 14 ze str1
49						2	1	č. 39 a 15 ze str1
50						2	0	č. 39 a 16 ze str1

Tabulka 1: Výsledky zápasů před a po analýze cyklických pravidel

6.1.1 Hodnocení výsledků

Z tabulky č. 1 je zřejmé, že ne vždy se optimalizace s cyklickými pravidly projevila jako účinná. Nicméně zajímavým poznatkem je, že ani v jednom zápasu se tato optimalizace neprojevila jako negativní. Můžeme tvrdit, že ikdyž tuto optimalizaci aplikujeme, tak se při nejhorším dočkáme výsledku stejného jako bez optimalizace, ale ne horšího.

Pozitivně se optimalizace projevila u 20 z 50 zápasů, takže úspěšnost byla 40%. Prozkoumali jsme logovací soubory původních her bez optimalizace. V případě první dvojice strategií (str1 a str3) se v průběhu původní hry využila pravidla z detekovaného cyklu č. 12 a 14, ale i 15. Ostatní pravidla z cyklů se během hry nevyužila. Při odstranění pravidla č. 15 se sice výsledek zápasu nezměnil, ale to přisuzujeme faktu, že se využilo jen ve třech zápasech hry v logovacím souboru (z celkových 1200 řádků). Pravidla č. 12 a 14 byla během hry využita častěji a proto při jejich odstranění došlo ke změně postupu týmu a jinému, lepšímu výsledku. Dalo by se říci, že se tým nedostával do zacyklení a tudíž byl úspěšnější. U dvojice strategií str1 a str5 je zřejmé, že úspěšnější bylo odstranění pravidla č. 7 z cyklu. Pravidlo č. 7 mělo větší podíl na hře než pravidlo č. 39, proto jeho odstranění mělo také větší efekt.

Ověřili jsme, že na tomto výše uvedeném principu se projevila optimalizace i u ostatních úspěšných zápasů v tabulce. Ikdyž například u hry strategií str15 a str2 se obě pravidla z cyklu (8 a 10) objevují v logu původní hry, ale úspěšnost se projevila jen při odstranění pravidla č. 8. Tento fakt vysvětlujeme tím, že pravidlo č. 8 se použilo hned na začátku hry, kdežto pravidlo č. 10 až téměř v závěru a proto jeho odstranění nemělo tak zásadní vliv na průběh hry.

6.2 Využití detekce hub pravidel

Ze sady strategií jsme vybrali ty, u kterých bylo detekováno nějaké hub pravidlo, abychom mohli opět výsledky zápasů porovnat před a po optimalizaci strategie. V tabulce č. 2 můžeme vidět výsledky vybraných zápasů. Se stejnou dvojicí strategií se vždy odehrál zápas před odstraněním hub pravidla a po odstranění, resp. úpravě hub pravidla.

V tabulce č. 2 jsou červeným textem značeny zápasy, které se projevily po optimalizaci jako úspěšnější. Řádky s zápasy, které mají šedé pozadí značí negativní vliv na konečný výsledek zápasu po optimalizaci.

6.2.1 Hodnocení experimentu

V tabulce č. 2 vidíme výsledky 51 testovacích zápasů. Je zřejmé, že úspěšnost nebo selhání detekce hub pravidel není vůbec jednoznačná.

Pomocí hub detekce jsme určili 3 různé strategie, u kterých se hub pravidla objevila. V případě strategie str1, str4 a str15 se jednalo o okrajová hub pravidla, tzn. že úprava strategie spočívala jen v jejich odstranění. U strategie str18 byly nalezeny dvě hub pravidla - pravidlo č. 4 a 8. Pravidlo č. 8 je také okrajové, takže bylo jen odstraněno. Nicméně pravidlo č. 4 se nacházelo uprostřed strategie po jeho odstranění následovala automatická úprava dalších pravidel.

Zápasy bez optimalizace						Zápasy po hub analýze			
#	Náš tým	Soupeř	Scóre		Hub	Náš tým	Soupeř	Scóre	
1	str4	str1	0	0	1	str4_NEW	str1	0	0
2	str4	str3	0	0	1	str4_NEW	str3	0	0
3	str4	str5	2	0	1	str4_NEW	str5	2	0
4	str4	str7	0	0	1	str4_NEW	str7	0	0
5	str4	str9	0	0	1	str4_NEW	str9	0	0
6	str4	str11	0	0	1	str4_NEW	str11	0	0
7	str4	str13	0	0	1	str4_NEW	str13	0	1
8	str4	str15	0	0	1	str4_NEW	str15	0	0
9	str4	str17	0	0	1	str4_NEW	str17	0	0
10	str5	str2	0	2	1	str5_NEW	str2	0	2
11	str5	str4	0	0	1	str5_NEW	str4	0	0
12	str5	str6	0	1	1	str5_NEW	str6	0	1
13	str5	str8	1	1	1	str5_NEW	str8	1	1
14	str5	str10	0	0	1	str5_NEW	str10	0	0
15	str5	str12	0	0	1	str5_NEW	str12	0	0
16	str5	str14	1	0	1	str5_NEW	str14	1	0
17	str5	str16	0	1	1	str5_NEW	str16	0	1
18	str15	str1	0	0	8	str15_NEW	str1	0	0
19	str15	str2	0	0	8	str15_NEW	str2	1	0
20	str15	str3	0	0	8	str15_NEW	str3	1	0
21	str15	str4	0	0	8	str15_NEW	str4	0	0
22	str15	str5	0	0	8	str15_NEW	str5	0	1
23	str15	str6	0	0	8	str15_NEW	str6	2	0
24	str15	str7	0	1	8	str15_NEW	str7	0	1
25	str15	str8	0	0	8	str15_NEW	str8	0	0
26	str15	str9	0	0	8	str15_NEW	str9	0	1
27	str15	str10	0	1	8	str15_NEW	str10	0	2
28	str15	str11	0	1	8	str15_NEW	str11	2	0
29	str15	str12	0	0	8	str15_NEW	str12	0	0
30	str15	str13	0	1	8	str15_NEW	str13	0	0
31	str15	str14	0	0	8	str15_NEW	str14	0	0
32	str15	str16	1	2	8	str15_NEW	str16	0	0
33	str15	str17	0	0	8	str15_NEW	str15	0	0
34	str15	str18	0	0	8	str15_NEW	str16	0	0
35	str18	str1	0	0	4, 8	str18_NEW	str1	0	1
36	str18	str2	1	1	4, 8	str18_NEW	str2	1	2
37	str18	str3	2	1	4, 8	str18_NEW	str3	0	0
38	str18	str4	1	1	4, 8	str18_NEW	str4	0	0
39	str18	str5	0	0	4, 8	str18_NEW	str5	0	0
40	str18	str6	0	1	4, 8	str18_NEW	str6	0	0
41	str18	str7	0	0	4, 8	str18_NEW	str7	2	2
42	str18	str8	0	0	4, 8	str18_NEW	str8	1	0
43	str18	str9	0	0	4, 8	str18_NEW	str9	0	0
44	str18	str10	0	0	4, 8	str18_NEW	str10	3	2
45	str18	str11	0	0	4, 8	str18_NEW	str11	1	1
46	str18	str12	0	0	4, 8	str18_NEW	str12	1	0
47	str18	str13	0	0	4, 8	str18_NEW	str13	0	0
48	str18	str14	0	0	4, 8	str18_NEW	str14	0	1
49	str18	str15	0	0	4, 8	str18_NEW	str15	0	0
50	str18	str16	0	0	4, 8	str18_NEW	str16	1	0
51	str18	str17	0	0	4, 8	str18_NEW	str17	1	0

Tabulka 2: Výsledky zápasů před a po analýze hub pravidel

Pouze 12 zápasů se po hub optimalizaci projevilo s lepším výsledkem, tudíž pozitivním efektem hub detekce. Můžeme tvrdit, že úspěšnost detekce byla přibližně 23,5%.

Na druhou stranu se u této optimalizace objevily i zápasy, na které měla hub detekce negativní dopad a výsledek se zhoršil. Takových zápasů jsme zaznamenali 7, což je přibližně 13,7%. Po prozkoumání logovacích souborů zápasů lze tvrdit, že negativní efekt hub optimalizace spočívá v prostém odstranění detekovaného pravidla. Ve všech negativních hrách se toto pravidlo v původních neupravených hrách podílelo na průběhu hry v různé míře.

Detekovaná hub pravidla se podílí na hře v původních zápasech před optimalizací, ale i v případech pozitivního dopadu hub detekce, čili podle tohoto faktu nelze nic konkrétního usuzovat.

V obecné rovině nelze s určitostí tvrdit, že hub optimalizace je účinná. Ve většině případů se buď na výsledku neprojeví vůbec nebo se výsledek zlepší. Můžou se ovšem objevit situace, kdy je hub detekce na škodu.

6.3 Experimenty s generováním ofenzivních pravidel

Pro generování útočných pravidel jsme vybrali dvojice strategií takové, kde náš tým od soupeře dostal alespoň jeden gól. Generování útočných pravidel bychom nemohli využít bez soupeřových gólů. Opět jsme odehráli základní zápas bez optimalizací a zásahu do strategií. Následně jsme v průběhu hry využili generování útočných pravidel a přidávali jsme do naší strategie gólová pravidla soupeře. Takových zápasů jsme odehráli tolik, dokud docházelo ke změnám (přidání pravidla) ve strategii. Jakmile už bylo zřejmé, že se žádná nová pravidla nepřidávají, tak jsme testovací zápasy ukončili, protože je jasné, že ani výsledky už by se neměnily.

V tabulce č. 3 můžeme vidět výsledky experimentální zápasů. Na levé straně vidíme výsledky původních strategií bez úprav a v pravé části jsou výsledky dalších zápasů za využití generování útočných pravidel. Zápasy jsou označeny písmenem Z a pořadovým číslem zápasu (např.: druhý zápas = Z2). V posledním sloupci je uvedeno číslo pravidla ze soupeřovy strategie, které bylo přidáno do naší strategie. Následuje závorka s číslem zápasu, ve kterém přidání proběhlo.

6.3.1 Hodnocení experimentu

Pro testování generování útočných pravidel byly odehrány zápasy s 50 různými dvojicemi strategií. Zápasy se hrály tak dlouho, dokud se již útočná pravidla negenerovala. Za úspěšnou sérii her budeme považovat takové hry, ve kterých se výsledek zápasu oproti původní hře zlepšil - tzn. náš tým vyhrál, snížil se počet obdržných gólů nebo došlo k remíze a zvýšil se počet námi vstřelených gólů oproti původní hře.

Pro 20 dvojic strategií se generování útočných pravidel projevilo jako úspěšné a výsledek se zlepšil. Negativní dopad jsme zaznamenali pouze v 5 případech. V ostatních hrách se generování útočných pravidel nijak neprojevilo a výsledek se nezměnil. Čili úspěšnost této metody je 40%, negativní dopad byl zaznamenán v 10% případů.

Zápasy bez úprav				Zápasy při využití generování ofenzivních prav.			
#	Náš tým	Soupeř	Scóre	Z1	Z2	Z3	Přidané pravidlo (zápas)
1	str1	str15	0:1	0:1			žádné, pr. č. 4 již str1 obsahuje
2	str5	str2	0:2	0:1	0:1		č. 10 (Z1)
3	str5	str3	1:1	1:1	1:1		č. 6 (Z1)
4	str5	str6	0:1	0:2	0:2		č. 3 (Z1)
5	str5	str7	0:1	2:2	1:1		č. 4 (Z1)
6	str5	str8	1:1	1:1	1:1	0:0	č. 1 (Z1), č. 5 (Z2)
7	str5	str16	0:1	0:1	1:0		č. 5 (Z1)
8	str1	str3	1:1	1:1	1:1		č. 21 (Z1)
9	str1	str5	0:2	1:1	0:2	0:2	č. 31 (Z1), č. 2 (Z2)
10	str15	str7	0:1	0:1	0:1		č. 2 (Z1)
11	str15	str10	0:1	0:1	0:1		č. 8 (Z1)
12	str15	str11	0:1	0:2	0:0		č. 3 (Z1)
13	str15	str13	0:1	0:1	0:1		č. 8 (Z1)
14	str15	str16	1:2	0:2	0:0		č. 3 (Z1)
15	str18	str2	1:1	1:1	1:1		č. 13 (Z1)
16	str18	str3	2:1	2:1	1:0		č. 16 (Z1)
17	str18	str4	1:1	2:2	1:0		č. 8 a č. 13(Z1)
18	str18	str6	0:1	0:1	0:1		č. 2 (Z1)
19	str2	str7	0:2	0:1	1:0		č. 4 (Z1)
20	str2	str10	0:1	0:1	0:1		č. 8 (Z1)
21	str2	str11	0:2	0:1	0:1		č. 5 (Z1)
22	str2	str12	0:1	0:2	2:0		č. 1 a č. 3(Z1)
23	str2	str13	1:1	0:2	0:1		č. 8 (Z1)
24	str2	str14	0:1	0:2	0:1		č. 8 (Z1)
25	str2	str17	1:1	0:2	1:0		č. 1 a č. 3(Z1)
26	str3	str2	0:1	0:2	0:1		č. 13 a č. 15(Z1)
27	str3	str6	0:1	1:1	1:0		č. 1 (Z1)
28	str3	str7	0:2	0:1	0:0		č. 1 (Z1)
29	str3	str8	1:2	0:1	1:1		č. 1 (Z1)
30	str3	str9	1:1	1:1	1:1		č. 1 (Z1)
31	str3	str10	1:1	1:2	1:1		č. 5 a č. 1(Z1)
32	str3	str11	2:1	0:1	2:2	0:2	č. 3 (Z1), č. 8(Z2)
33	str3	str14	0:1	0:1	0:1		č. 15(Z1)
34	str3	str16	1:1	0:1	1:1		č. 3 (Z1)
35	str4	str8	0:1	1:3	0:1		č. 5 (Z1)
36	str4	str10	1:1	1:1	0:1		č. 8(Z1)
37	str4	str16	0:1	0:1	0:0		č. 1 (Z1)
38	str6	str7	0:1	1:1	0:1		č. 4 (Z1)
39	str6	str8	0:1	0:1	0:0		č. 4 (Z1)
40	str6	str9	0:1	2:2	1:0		č. 8 (Z1)
41	str6	str17	1:1	1:1	1:1		č. 3 (Z1)
42	str7	str2	1:1	1:1			žádné, pr. č. 15 již str7 obsahuje
43	str7	str3	0:1	0:1	2:2	0:0	č. 18 (Z1), č. 16 a č. 21 (Z2)
44	str7	str10	0:1	1:1	0:1	0:1	č. 5 (Z1), č. 1 (Z2)
45	str7	str12	0:1	0:1	0:1		č. 1 (Z1)
46	str7	str16	0:1	0:1	2:2	1:1	č. 1 (Z1), č. 5 a č. 16 (Z2)
47	str7	str17	0:1	0:1	0:2		č. 1 (Z1)
48	str8	str2	0:1	0:1	0:1		č. 13 (Z1)
49	str8	str7	0:1	0:1	0:1		č. 2 (Z1)
50	str8	str13	0:1	0:1	0:0		č. 5 (Z1)

Tabulka 3: Výsledky zápasů před a po využití generování ofenzivních pravidel

Negativní výsledek hry je ve 3 případech důsledek přidání pravidla, které hru ovlivnilo tak, že se roboti dostali do situace, ve které náš tým dostal gól. Ve zbylých 2 případech došlo ke kolizi ve hře, kde se hráči dlouhý čas „přetlačovali“ o míč. Tento typ kolizí není prozatím v simulátoru dořešen a výsledky takových her bychom neměli považovat za relevantní. Proto negativní vliv generování útočných pravidel se fakticky projevil jen u 6% her.

6.4 Experimenty s kombinací optimalizací

Jako poslední experiment jsme se rozhodli využít kombinaci dvou optimalizací, které se jevily jako účinné. Nejdříve jsme na strategii aplikovali optimalizaci pomocí detekce cyklických pravidel a následně jsme na vhodné výsledky použili generování útočných pravidel.

V tabulce č. 4 jsou výsledky testovacích zápasů při kombinování optimalizací. V levé části jsou uvedeny názvy strategií a výsledek hry bez jakéhokoliv zásahu. V další sloupci jsou uvedeny cykly pravidel, které byly v naší strategii detekovány. V následujících dvou sloupcích uvádíme skóre zápasu po odstranění určitých pravidel ze strategie našeho týmu (vlevo) tak, aby se již netvořily cykly. Další sloupec obsahuje čísla pravidel, která byla z naší (levé) strategie odstraněna. V posledních třech sloupcích tabulky jsou uvedeny výsledky zápasů při generování útočných pravidel na strategii, která již byla optimalizována od cyklických pravidel.

Generování útočných pravidel jsme aplikovali tak dlouho dokud se naše strategie měnila a přibývaly v ní nová pravidla. Poslední skóre zápasu je vždy zápas během kterého jsme žádné nové pravidlo již nepřidali.

Pro tento experiment jsme vybrali takové zápasy, ve kterých náš tým i po detekci cyklických pravidel dostal alespoň jeden gól. To proto, abychom mohli aplikovat a porovnat účinnost generování útočných strategií.

6.4.1 Hodnocení experimentu

Pro otestování úspěšnosti kombinací optimalizací jsme využili detekci cyklických pravidel a generování útočných strategií. Odehráli jsme 49 různých zápasů s různými dvojicemi strategií. Situace, kdy se kombinace optimalizací projevila pozitivně jsou v tabulce č. 4 zvýrazněny červeným textem. Je zřejmé na první pohled, že se tato kombinace optimalizací neprojevila jako úspěšná u mnoha zápasů. Pozitivní výsledek jsme zaznamenali jen u 4 zápasů, což tvoří přibližně 8,2%. U ostatních her se výsledek ani nezlepšil, ani nezhoršil.

Z toho experimentu můžeme vyvodit, že zmíněné dvě optimalizace lze kombinovat bez toho, aniž by se u naší strategie výsledek zápasu někdy zhoršil. Buď se výsledek zlepšil nebo se optimalizace nijak neprojevila.

Základní zápasy bez optimalizace					Zápasy po analýze cyklů		Ofen. prav.		
#	Náš tým	Soupeř	Scóre	Cykly	Scóre	Odstraněné prav.	Z1	Z2	Z3
1	str1	str3	1:1	39-7	1:1	č. 7 a 11	1:1	1:1	
2					1:1	č. 7 a 15	1:1	1:1	
3					1:1	č. 7 a 16	1:1	1:1	
4					1:1	č. 39 a 11	1:1	1:1	
5					1:1	č. 39 a 15	1:1	1:1	
6					1:1	č. 39 a 16	1:1	1:1	
7	str1	str5	0:2	39-7 11-12- -14-15-16	0:1	č. 7 a 12	0:1	0:1	
8					0:2	č. 39 a 11	1:1	1:1	0:2
9					0:2	č. 39 a 12	1:1	0:2	0:2
10					0:2	č. 39 a 14	1:1	0:2	0:2
11					0:2	č. 39 a 15	1:1	0:2	0:2
12					0:2	č. 39 a 16	1:1	0:2	0:2
13	str5	str3	1:1	25-26	1:1	č. 25	1:1	1:1	
14					1:1	č. 26	1:1	1:1	
15	str5	str6	0:1	25-26	0:1	č. 25	0:2	0:1	
16					0:1	č. 26	0:2	0:1	
17	str5	str7	0:1	25-26	0:1	č. 25	2:2	1:1	
18					0:1	č. 26	2:2	1:1	
19	str5	str16	0:1	25-26	0:1	č. 25	0:1	1:0	
20					0:1	č. 26	0:1	1:0	
21	str15	str1	0:0	8-10	0:0	č. 10	1:1		
22	str15	str16	1:2	8-10	0:0	č. 10	0:1	1:0	
23	str1	str15	0:1	39-7 11-12- -14-15-16	0:1	č. 7 a 11	0:1		
24					2:1	č. 7 a 15	0:1	2:1	
25					0:1	č. 39 a 11	0:1		
26					2:1	č. 39 a 15	0:1	2:1	
27	str20	str5	0:2	9-35, 9-72 28-30, 28-67	0:2	č. 35, 72, 28	0:1	0:2	
28					0:2	č. 35, 72, 30, 67	0:1	0:2	
29	str21	str3	2:3	9-43, 61-63	2:3	č. 43 a 61	2:3	2:3	
30					2:3	č. 43 a 63	2:3	2:3	
31	str21	str5	0:1	9-43, 61-63	0:1	č. 43 a 61	0:1	0:1	
32					0:1	č. 43 a 63	0:1	0:1	
33	str19	str7	0:2	13-45, 13-74 44-45, 44-74 38-40-45	1:2	č. 13, 44 a 38	1:2	0:0	
34					1:2	č. 13, 44 a 45	1:2	0:0	
35					0:2	č. 45 a 74	1:2	0:0	
36	str21	str8	2:1	9-43, 61-63	2:1	č. 9 a 61	2:1	2:1	
37					2:1	č. 9 a 63	2:1	2:1	
38					2:1	č. 43 a 61	2:1	2:1	
39					2:1	č. 43 a 63	2:1	2:1	
40	str20	str3	1:1	9-35, 9-72 28-30, 28-67	1:1	č. 9 a 28	1:1	1:1	
41					1:1	č. 9, 30 a 67	1:1	1:1	
42					1:1	č. 35, 72 a 28	1:1	1:1	
43					1:1	č. 35, 72, 30, 67	1:1	1:1	
44	str22	str8	1:1	7-27, 20-22 20-59, 27-36 1-3-27	1:1	č. 27 a 20	0:1	1:1	
45					1:1	č. 27, 22 a 59	1:1	1:1	
46					1:1	č. 7, 20, 36, 1	0:1	1:1	
47					1:1	č. 3, 20 a 27	0:1	1:1	
48	str22	str7	1:2		1:1	č. 59, 20 a 27	0:1	1:1	
49					2:1	č. 27, 22 a 59	1:2	2:1	

Tabulka 4: Výsledky zápasů s kombinací optimalizací

7 Závěr

V práci jsme představili hru fotbalu robotů a její význam v rámci výzkumu a vývoje umělé inteligence. Krátce jsme se věnovali soutěží RoboCup a MiroSot a vlastnostem, jaké by měl fotbalista robot mít.

Dále jsme se věnovali simulátoru robotů, který je vyvíjen na katedře informatiky. Určili jsme si pojmy, které jsou v práci použity v souvislosti s fotbalem robotů, strategiemi a simulací zápasů. Popsali jsme také aktuální stav implementace, ve které se simulátor nacházel před začátkem této práce. Vzhledem k tomu, že se naše práce výhradně zabývala strategiemi, tak jsme také popsali jakým způsobem a pomocí kterých nástrojů je možné strategii fotbalového týmu v simulátoru vytvořit a aplikovat do hry.

Dále jsme definovali speciální typy pravidel ve strategii, která se zdála být důležitější než ostatní pravidla. Jako speciální pravidla jsme stanovili hub, cyklická a útočná pravidla. Teoreticky jsme navrhli algoritmy pro jejich detekci a následné využití. Struktura těchto algoritmů je v práci popsána do detailu.

Navržené algoritmy jsme prakticky naimplementovali do simulátoru fotbalu robotů. Vytvořili jsme novou třídu `SG.StrategyOptimization`, do které jsme zapracovali většinu kódu pro detekci a využití hub pravidel, cyklických pravidel a útočných pravidel. V rámci implementace jsme také upravili GUI tak, aby bylo možno ve `StrategyCreatoru` jednotlivé optimalizace využít. V práci jsou stručně představeny některé metody třídy `SG.StrategyOptimization`.

Po ukončení praktické implementace jsme provedli experimenty pro ověření teorií týkajících se speciálních pravidel ve strategiích. Pro každé ze tří speciálních pravidel jsme odehráli okolo 50 testovacích zápasů. Využili jsme k tomu 22 různých strategií. Po experimentech se nejlépe projevila detekce na cyklická pravidla, při které se vždy výsledek buď nezměnil nebo zlepšil. Hub detekce se projevila jako nejednoznačná, protože u některých zápasů se projevuje pozitivně, ale u jiných strategií působí negativně a výsledek zhoršuje. Generování útočných pravidel během her se ukázalo úspěšné ve 40% zápasů. Negativní působení se také projevilo, ale v minimálním počtu her. Posledním experimentem byla kombinace detekce cyklických pravidel a generování útočných pravidel na stejných strategiích. Ukázalo se, že tyto dvě optimalizace lze kombinovat bez toho, aniž by se výsledek hry zhoršil. Úspěšnost byla sice nízká (8,2%), ale negativně se tato kombinace neprojevila u žádné z her.

Implementace simulátoru pořád ještě není dokonalá a jak již bylo řečeno, simulátor se stále vyvíjí. Proto jsme se samozřejmě potkali s několika problémy. Jedním z nich je řešení kolizí při hře, které sice již v simulátoru implementováno je, ale ne pro všechny možné situace. Při hře se stává, že se roboti nekonečně přetlačují o míč nebo že se zaseknou v rohu hřiště u mantinelu a s míčem již nepohnou. Ve hře také chybí automatický rozhodčí, který by rozhodl o postavení hráčů po daných herních situacích (např., že po obdržení gólu vykopává brankář apod.). Tyto nedostatky mohou vést k tomu, že některé výsledky experimentálních her nejsou do jisté míry nerelevantní. Námětem na další práci je otestování optimalizací strategií po úplném dokončení implementace simulátoru a odstranění jeho nedostatků.

Zajímavým námětem na další řešení může také být otestování optimalizace strategií s různými gridy, větším počtem hráčů na hřišti a případném přidání možnosti hrát tzv. power play, tzn. že se brankář může účastnit hry i mimo bránu.

8 Literatura

- [1] Sammut C. *Robot soccer*, Wiley Interdisciplinary Reviews: Cognitive Science, Volume 1, Issue 6, pages 824–833, November/December 2010.
- [2] Jie Wu., Ochodková E., Martinovič J., Snášel V., Abraham A. *Analysis of Loop Strategies in Robot Soccer Game, Intelligent Systems Design and Applications (ISDA)*, Cordoba, Spain, 2011, ISSN 2164-7143, ISBN 978-1-4577-1676-8.
- [3] Jie Wu., Svatoň V., Martinovič J., Ochodková E., Snášel V. *Generation of Offensive Strategies in Robot Soccer Game, Intelligent Networking and Collaborative Systems (IN-CoS)*, Fukuoka, Japan, 2011, ISBN 978-1-4577-1908-0.
- [4] Jie Wu., Snášel V., Ochodková E., Martinovič J., Svatoň V., Abraham A. *Improvement of Hub Strategy in Robot Soccer Game, Soft Computing and Pattern Recognition (SoCPaR)*, Dalian, China, 2011, ISBN 978-1-4577-1195-4.
- [5] Žořtá L. *2D simulátor fotbalu robotů*, Ostrava, 2009, Diplomová práce.
- [6] Balon S. *Strategie a taktiky pro fotbal robotů*, Ostrava, 2008, Bakalářská práce.
- [7] Karel Čapek [online]. Dostupný z WWW:
http://cs.wikipedia.org/wiki/Karel_%C4%8Capek [cit. březen 2012]
- [8] Robocup [online]. Dostupný z WWW:
<http://www.robocup.org> [cit. březen 2012]
- [9] Robotický fotbal pro zábavu, výuku i vědeckou práci [online]. Dostupný z WWW:
<http://www.automatizace.cz/article.php?a=684> [cit. březen 2012]
- [10] RoboCup German Open 2010: B-Human Team qualifiziert sich für WM in Singapur [online]. Dostupný z WWW: <http://blog.joergboesche.de/themen/misc/page/2> [cit. březen 2012]
- [11] Strategie [online]. Dostupný z WWW: <http://cs.wikipedia.org/wiki/Strategie> [cit. březen 2012]

A Obsah CD

- text práce
- zdrojové soubory se strategiemi k testovacím hrám
- zdrojové soubory - 2D simulátor fotbalu robotů s implementací optimalizací strategií